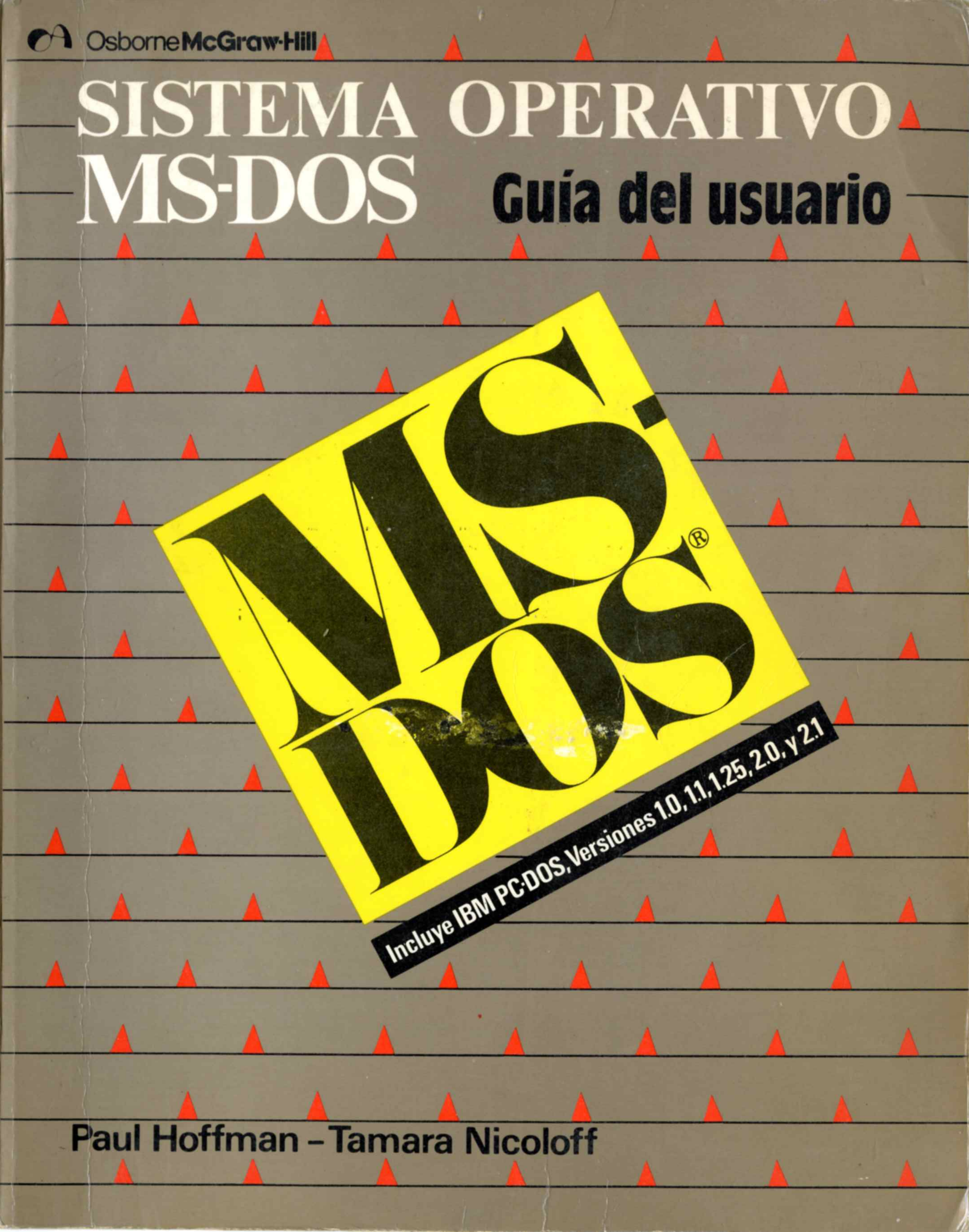


Osborne McGraw-Hill
SISTEMA OPERATIVO
MS-DOS
Guía del usuario
MS-DOS®
Incluye IBM PC-DOS, Versiones 1.0, 1.1, 1.25, 2.0, y 2.1
Paul Hoffman – Tamara Nicoloff

SISTEMA OPERATIVO
MS-DOS
Guía del usuario

SISTEMA OPERATIVO

MS-DOS
Guía del usuario

Paul Hoffman
Tamara Nicoloff

Traducción
Carmen Fernández Chamizo
Doctor en Ciencias Físicas
Departamento de Informática y Automática
Facultad de Ciencias Físicas
Universidad Complutense de Madrid

Revisión técnica
Antonio Vaquero Sánchez
Catedrático de Informática
Facultad de Ciencias Físicas
Universidad Complutense de Madrid

OSBORNE/McGraw-Hill

MADRID • BOGOTA • BUENOS AIRES • GUATEMALA • LISBOA • MEXICO • NUEVA YORK
PANAMA • SAN JUAN • SANTIAGO • SÃO PAULO
AUCKLAND • HAMBURGO • JOHANNESBURGO • LONDRES • MONTREAL • NUEVA DELHI
PARIS • SAN FRANCISCO • SINGAPUR • ST. LOUIS • SIDNEY • TOKIO • TORONTO

SISTEMA OPERATIVO MS-DOS. Guía del usuario

Atlacomulco 499-501, Naucalpan de Juárez, Edo. de México
Miembro de la Cámara Nacional de la Industria Editorial, Reg. Núm. 465

ISBN 968-451-762-9

Traducido de la primera edición en inglés de

MS-DOS USER'S GUIDE

ISBN 0-88134-131-2

Edición exclusiva para Ediciones La Colina, S. A. (España)

ISBN: 84-7615-026-1
Depósito legal: M. 12895-1986

Compuesto en FER Fotocomposición, S. A. Lenguas, 8. 28021 Madrid
Impresión: Artes Gráficas EMA, S. A. Miguel Yuste, 27. 28037 Madrid

PRINTED IN SPAIN - IMPRESO EN ESPAÑA

INDICE DE MATERIAS

Introducción		vii
Capítulo 1	Introducción al MS-DOS	1
Capítulo 2	Manejo del MS-DOS: Introducción a las órdenes y los ficheros	15
Capítulo 3	Ordenes del MS-DOS	45
Capítulo 4	Uso avanzado del MS-DOS	159
Capítulo 5	Tratamiento de incidentes	185
Capítulo 6	Programas de aplicaciones para el MS-DOS	199
Capítulo 7	Otras publicaciones sobre el MS-DOS	257
Apéndice A	Listas de referencia de órdenes	267
Apéndice B	Diferencias entre las versiones 1 y 2 del MS-DOS	273
Apéndice C	Compatibilidad con el IBM PC	277
Apéndice D	Computadoras que ejecutan el MS-DOS	279
Apéndice E	Fabricantes de hardware y software	291
Apéndice F	Glosario	299
Indice		309

AGRADECIMIENTOS

Pam Edstrom, Chris Larson y Amy Powers, de Microsoft Corporation nos ayudaron en la investigación sobre el MS-DOS. Dale Gifford nos dio excelentes indicaciones técnicas. Jack Krasner nos aconsejó sobre los paquetes contables. Bob Kermish, de Computerland en Oakland, CA, contestó muchas preguntas acerca de IBM. Bonnie Bozorg y Karen Nebelkopf nos ayudaron con el procesamiento de texto. Les damos las gracias a todos.

Paul Hoffman
Tamara Nicoloff

INTRODUCCION

Este libro será útil para cualquier persona que posea una microcomputadora con el sistema operativo MS-DOS, tal como el IBM PC o el PC «junior». Contiene información de lo que es el MS-DOS, lo que se puede hacer con él, cómo darle órdenes y qué programas de aplicación (tales como procesadores de texto y paquetes de contabilidad) se pueden usar con él.

Este libro responde a las preguntas que hacen la mayor parte de los usuarios del MS-DOS: ¿Qué es el MS-DOS? ¿Qué puede hacer por mí? ¿Necesito pasar un montón de tiempo aprendiendo el MS-DOS? ¿Qué software puedo usar con mi computadora MS-DOS?

Aunque este libro está escrito fundamentalmente de forma que los usuarios principiantes puedan entenderlo fácilmente, contiene también una gran cantidad de informaciones útiles para usuarios avanzados. Si su manual le ha dejado desorientado, este manual le ayudará a entender muchos de los conceptos expuestos.

Algunos fabricantes han modificado el MS-DOS para adaptarlo a sus computadoras y otros incluso le han dado un nuevo nombre (tal como PC-DOS para el IBM PC). Sin embargo, estas variaciones del MS-DOS son usualmente de poca importancia. Este libro sirve para todas las variaciones del MS-DOS en las versiones 1 y 2.

¿Por qué debería usted leer este libro?

Lo mismo si se trata de nuevos usuarios como si se trata de usuarios que llevan utilizando la computadora durante años, la mayoría no están familiarizados con todas las funciones que su sistema operativo MS-DOS puede realizar. Este libro proporciona una introducción lenta pero directa a estas funciones. El libro también ayuda a juzgar hasta qué punto son apropiadas las características del MS-DOS para sus propias necesidades. El libro tiene otras importantes características entre las que se incluyen:

- **Lista completa de todas las órdenes del MS-DOS.** Todas las órdenes del MS-DOS se presentan en el capítulo 3. Están organizadas en grupos funcionales, de manera que se puede determinar rápidamente la relación entre las órdenes. Se dan muchos ejemplos de cada orden para mostrar su uso en una gama de situaciones.
- **Numerosos ejemplos claros y extensos.** Cada ejemplo muestra la utilización práctica de los conceptos importantes del MS-DOS.
- **Un capítulo sobre cómo tratar los incidentes.** El capítulo 5 muestra qué hacer cuando un disco se ha «caído» (crashed), cuando hay un corte de

energía eléctrica o cuando los ficheros se borran accidentalmente; también se dan consejos para evitar errores graves.

- **Guías detalladas para encontrar programas de aplicación, revistas y libros.** Los capítulos 6 y 7 proporcionan algo más que simples referencias. Dan también consejos de cómo y dónde comprar y se describen métodos fáciles para mantenerse al día a pesar de los rápidos cambios en el mercado.
- **Una actualizada descripción del MS-DOS.** Este libro informa de las versiones 1 y 2 del MS-DOS. Si usted tiene la versión 1, que fue lanzada con anterioridad y que tiene menos órdenes, no se preocupe de si va a poder seguir las instrucciones. El libro siempre indica qué órdenes o comentarios son aplicables exclusivamente a la versión 2.
- **Explicaciones no técnicas de las características avanzadas.** El capítulo 4 sobre las aplicaciones avanzadas del MS-DOS le ayudará a entender por qué el MS-DOS es un sistema operativo superior. Para minimizar la confusión, las características avanzadas se describen después de las órdenes más comunes del MS-DOS.

Este libro difiere en varios aspectos importantes de otros libros sobre el mismo tema. En primer lugar, no asume que el manual del MS-DOS, que viene con su computadora, es una guía suficiente de las órdenes MS-DOS. Este libro llena los huecos del manual, proporcionando explicaciones completas. Tampoco insiste en que los usuarios del MS-DOS necesiten aprender a programar en BASIC. En cambio, este libro indica cómo comprar los programas ya existentes que pueden hacer casi todo.

Tercero, este libro no sugiere que usted necesita entender todas las características y las órdenes del MS-DOS. Al contrario, usted probablemente sólo usará regularmente alrededor de diez órdenes, durante toda la vida de su computadora. Si algunas características no le sirven para nada, el libro lo hace constar claramente.

Y finalmente, este libro no asume que usted posee un IBM PC. Cualquiera que sea la computadora MS-DOS que usted tiene, este libro le será de utilidad. Aunque la versión IBM PC del MS-DOS se explica en profundidad, las computadoras MS-DOS más populares tienen características muy similares en sus versiones del MS-DOS.

¿Cómo está organizado este libro?

Los autores recuerdan intensamente lo duro que resulta aprender todo lo relativo a las computadoras y lo frustrante que pueden ser las primeras experiencias sin una buena guía. Este libro está lleno de consejos prácticos, no solamente descripciones del sistema operativo. El estilo del libro y su organi-

zación permiten un rápido y fácil entendimiento, de forma que usted pueda usar las órdenes del MS-DOS lo más pronto posible.

Los primeros dos capítulos presentan la base necesaria para iniciarse en el MS-DOS; el resto del libro muestra métodos específicos para trabajar con el MS-DOS. Para sacar el mayor partido de este libro, debería leer los tres primeros capítulos y por lo menos hojear el resto. Los usuarios más avanzados pueden examinar superficialmente la mayor parte de los dos primeros capítulos y concentrarse en las órdenes propiamente dichas; se debería leer al menos el comienzo de cada capítulo para tener una idea de toda la información que se presenta.

Debido a que las descripciones son extensas y completas, se han utilizado algunos términos técnicos, cuando el lenguaje común no resulta suficiente. Sin embargo, todos los términos técnicos del libro están en letra cursiva, se definen la primera vez que aparecen y se usan consecuentemente después. Estos términos aparecen también listados en el glosario. A continuación, se resumen los contenidos de cada capítulo y apéndice.

El capítulo 1 presenta una introducción al sistema operativo MS-DOS, que incluye las respuestas a preguntas comunes tales como «¿Qué hace un sistema operativo?» y «¿Cómo se trabaja con el MS-DOS?» También describe cómo maneja el MS-DOS los distintos elementos hardware del sistema. El capítulo está dirigido a los usuarios principiantes, pero también es útil si usted ha usado otros sistemas operativos, tales como CP/M, pero está empezando con el MS-DOS.

El capítulo 2 muestra cómo se usan normalmente las órdenes del MS-DOS. Explica lo básico de la ejecución del MS-DOS e introduce en el uso de ficheros, directorios y ficheros para tratamiento por lotes. Una sesión de muestra, al final del capítulo, da un ejemplo de cómo se pueden utilizar los nuevos conocimientos.

La mayor parte de su interacción con la computadora será a través de las órdenes del MS-DOS. El capítulo 3 presenta una descripción completa de estas órdenes. Las órdenes están agrupadas por su función (por ejemplo, un grupo de órdenes que muestran los contenidos de los ficheros), presentando primero las órdenes usadas más frecuentemente. Los usuarios principiantes sólo necesitan leer los tres primeros grupos de funciones (mantenimiento de ficheros, salida de ficheros y mantenimiento de discos) para poder usar la mayor parte de las características del MS-DOS. El capítulo 3 es también una referencia que puede usarse cuando se requiere información específica.

El capítulo 4 presenta información avanzada del MS-DOS, incluyendo el directorio en forma de árbol de la versión 2. El capítulo es útil tanto para los usuarios avanzados del MS-DOS como para los principiantes con sistemas de disco duro. Sin embargo, los usuarios principiantes no necesitan leer este capítulo antes de usar el MS-DOS.

Cuando usted haya usado su computadora durante unas cuantas semanas, probablemente necesitará leer el capítulo 5, que habla de los incidentes, de cómo tratarlos y de cómo prevenirlos. También proporciona una lista de pro-

gramas y paquetes de aplicación disponibles para ayudar a la recuperación después de un error grave o para evitar que se produzcan dichos errores (como por ejemplo, los programas que hacen copias de los disquetes protegidos contra copia).

El capítulo 6 examina muchos de los programas de aplicaciones más populares y habla de cómo comprarlos. Cuanto más use su computadora, probablemente más querrá comprar software adaptado a sus necesidades particulares. El capítulo describe los tipos de programas disponibles, lista cientos de productos en todas las principales ramas del software y le dice cómo comprarlos.

El capítulo 7 habla de las revistas y libros que tratan del MS-DOS, de los productos MS-DOS y de computadoras en general. El capítulo abarca varias revistas populares y libros; también sugiere formas de ahorrar tiempo y dinero mientras se obtiene una colección de referencias.

El apéndice A contiene una corta guía alfabética de las órdenes del MS-DOS. El apéndice B describe las diferencias entre las versiones del MS-DOS. El apéndice C describe cómo determinar si una computadora es o no compatible con el IBM PC. El apéndice D le indica lo que debería buscar cuando compra una computadora; también proporciona información comparativa sobre más de 20 computadoras MS-DOS. El apéndice E lista los nombres y direcciones de los fabricantes de hardware y software mencionados a lo largo del libro.

Finalmente, el apéndice F es un glosario completo. Se puede usar como referencia rápida para las palabras o frases desconocidas, aunque la mayoría de los términos se definen en los capítulos, la primera vez que se usan.

Guía del usuario y manual de referencia

Como se ha mencionado anteriormente, este libro está diseñado para usarlo como introducción al MS-DOS y como manual de referencia. Si se usa este libro como referencia, el apéndice A porporciona un resumen alfabético de las órdenes, su sintaxis y la sección en la que están tratadas en el capítulo 3.

A lo largo del libro, usted encontrará temas que son importantes tanto para los principiantes como para los usuarios avanzados. Al comienzo de cada sección se indica a quién está dirigida la información, de forma que puede determinar qué material es importante para usted.

Mientras lea este libro, recuerde que el MS-DOS le proporciona un medio para comunicar con su computadora. Aunque tiene un gran número de características, usted sólo necesita conocer unas cuantas para utilizarlo de forma efectiva.

Esperamos que sus experiencias con el MS-DOS y con este libro tengan éxito.

PH
TN
1984

1

INTRODUCCION AL MS-DOS

Lo que hace el MS-DOS
Cómo usa el MS-DOS los discos
y los ficheros
El MS-DOS por dentro
Breve historia del MS-DOS
Lo que el MS-DOS no puede hacer

¿Por qué aprender MS-DOS? Si es usted como la mayoría de la gente, la razón principal para tener una computadora es el querer ejecutar *programas de aplicación,* tales como procesadores de texto y programas de contabilidad, que se adaptan para resolver problemas específicos o realizar ciertas tareas.

Los programas de aplicación son valiosos y este libro les dedica bastante atención en el capítulo 6. Sin embargo, los programas de aplicación no pueden hacerlo todo. Muchos de estos programas de aplicación asumen que usted ya sabe cómo usar las órdenes del MS-DOS, así que no incluyen instrucciones acerca de todas las funciones necesarias para dar soporte al programa. Por esta razón, usted puede encontrar que algunas de las tareas que quiere realizar (por ejemplo, formatear un disco nuevo para prepararlo para su uso) no están cubiertas por sus programas de aplicación. Estas tareas sólo puede hacerlas con su sistema operativo MS-DOS.

El formateado de discos es sólo una pequeña parte del papel del MS-DOS.

El sistema operativo MS-DOS es un programa que maneja el *hardware* de la computadora (incluyendo sus unidades de disco, teclado, monitor e impresora) y permite ejecutar otros programas. En realidad, usted no puede hacer mucho en su computadora sin un sistema operativo.

LO QUE HACE EL MS-DOS

Se puede imaginar que el MS-DOS es similar a la carlinga de un avión. Sin ella hay suficiente potencialidad, pero el avión no puede volar. El sistema operativo le permite a usted (el piloto) controlar su computadora diciéndole «dónde ir» y qué hacer. Igual que los mandos en una carlinga de avión, el sistema operativo coordina las partes de la computadora y proporciona un método fácil para que usted las controle. En este capítulo, usted comenzará a aprender cómo desempeña el MS-DOS este papel.

MS-DOS son las siglas *M*icro*S*oft *D*isk *O*perating *S*ystem; es un nombre genérico del sistema operativo patentado por Microsoft Corporation para su uso sobre varias microcomputadoras construidas por diversos fabricantes. (Algunos de estos fabricantes de computadoras han modificado el MS-DOS para sus computadoras y le han dado un nombre nuevo, tal como *PC-DOS* o *Z-DOS*).

Partiendo del nombre *sistema operativo de disco* (disk operating system, o *DOS*), se puede pensar que todo lo que hace el MS-DOS es manejar los discos. El término DOS ha permanecido en el vocabulario técnico porque hace muchos años los sistemas operativos hacían poco más que controlar discos. Sin embargo, el MS-DOS hace mucho más que eso. Proporciona un medio para decir a la computadora qué programa u orden se quiere ejecutar, dónde encontrará el programa o la orden y qué deberá hacer con ello. Por ejemplo; puede enviar información a la pantalla, a una impresora o a una puerta de comunicaciones para que se envíe a otro sistema. Se puede considerar que el sistema operativo trabaja en dos niveles.

El primer nivel es el de un sistema de gestión del hardware: El MS-DOS coordina la *unidad central de proceso* (*CPU*) de la computadora (que es el chip microprocesador que actúa como cerebro) con el resto del hardware de la computadora. En esta función de coordinación el MS-DOS toma el carácter que usted pulsa en el teclado, lo codifica en forma comprensible para la CPU y, a continuación, lo visualiza en el monitor en forma comprensible para usted. Por ejemplo, si usted está usando un programa de hoja de cálculo o un procesador de texto, el MS-DOS actúa como un intermediario que convierte las señales electrónicas, que genera el teclado, en códigos de control utilizables por el programa de aplicación. El MS-DOS también realiza pequeñas tareas relacionadas con el uso de los programas, tales como formatear un disco o indicar qué ficheros hay en un disco.

El segundo nivel en el que opera el MS-DOS es la función de utilidad. En este nivel, el MS-DOS ejecuta *órdenes* que permiten interactuar directamente con la computadora. Estas órdenes realizan funciones tales como nombrar los ficheros almacenados en disco o copiar ficheros de un disco a otro.

El MS-DOS trata a sus propias órdenes igual que a los programas de aplicación. Sin embargo, estas órdenes son más limitadas que la mayoría de los programas de aplicación. No realizan tareas como el procesamiento de textos o la contabilidad; en lugar de eso, se utilizan para mantener y gobernar la computadora. Cada orden tiene un nombre que suele ser fácil de recordar. Por ejemplo, para copiar información de un disco a otro se usa la orden COPY. Todas estas órdenes se tratan en el capítulo 3.

La mayor parte de este libro se dedica a este segundo nivel de utilidad del MS-DOS. Desde el capítulo 2 hasta el 5 se aprende a utilizar las órdenes del MS-DOS para el mantenimiento diario de ficheros y discos. Sin embargo, este capítulo estudia más detalladamente el primer nivel, aquel en el que el MS-DOS coordina todas las partes de la computadora. Los dos factores principales de este papel de coordinador son la comunicación entre los componentes hardware y la ejecución de programas de aplicación.

Comunicación con el hardware

La CPU de la computadora no puede funcionar bien sin un sistema operativo. Necesita un gestor maestro que controle el tráfico de información para poder coordinar toda la información obtenida del teclado, las unidades de disco y demás hardware de la computadora. El MS-DOS coordina el hardware y permite a la CPU comunicarse con casi cualquier otra parte de la computadora. Por ejemplo, sin el sistema operativo, la CPU no tiene forma de encontrar los datos y los programas en los discos.

Desde que usted carga el sistema operativo (usualmente cuando se enciende la computadora), éste se guarda en la memoria de acceso aleatorio (*RAM*). La RAM es igual que un enorme cuaderno de apuntes lleno de números e instrucciones; cuando un programa se ejecuta, lee algunos de los contenidos de RAM y cambia alguna información en ella. La memoria RAM es temporal; esto es, se mantiene exclusivamente gracias a la energía eléctrica de la computadora. Cuando se apaga la computadora, se pierde toda la información contenida en la RAM (es por esta razón por la que un corte de energía eléctrica puede ser desastroso cuando se está usando la computadora). Cualquier información en RAM se pierde en el momento en que hay un corte de energía, incluso si sólo dura un instante. Algunas computadoras MS-DOS tienen una batería de seguridad, que evita la pérdida de los datos en RAM, durante los cortes de energía.

Aunque la CPU no necesita ayuda del sistema operativo para comunicar con la RAM, es importante saber un poco acerca de la RAM, puesto que es donde permanecen todos los programas mientras se ejecutan. Se puede imagi-

nar la RAM como si fuera un conjunto de muchas cajas llamadas bytes. Cada *byte* es igual a la cantidad de memoria necesaria para almacenar un carácter simple del teclado (como una Q o una @). La cantidad de RAM que tiene una computadora se mide en *K* o *Kilobytes.* Un kilobyte es igual a 1.024 bytes.

Volviendo a la analogía de la RAM con un conjunto de cajas, se puede imaginar cada byte, o caja, dividida en ocho compartimentos, llamados bits, que pueden estar llenos o vacíos (ver figura 1-1). Como se puede ver, cada caja tiene un número, conocido como su *dirección,* que indica su posición en la RAM. La CPU constantemente usa estas direcciones para acceder a los contenidos de las cajas. La CPU también recibe datos de fuentes exteriores (tales como el disco o el teclado) y los coloca en RAM. Los contenidos de la RAM se modifican cuando se realizan tareas tales como cálculos y comparaciones entre números.

El proceso de introducir datos en RAM procedentes de fuentes como el teclado y, alternativamente, el tomar información de la RAM y enviarla a diferentes dispositivos hardware como la pantalla, se denomina *I/O* (input/output) o *entrada/salida.* Cada vez que se «lee» información de algún dispositivo hardware, como una unidad de disco, o se «escribe» información en un dispositivo hardware, como una impresora, el MS-DOS realiza I/O.

Un *dispositivo* es simplemente un componente hardware que usa I/O. Por ejemplo, una impresora es un dispositivo común que usted probablemente posee; otros dispositivos comunes son las unidades de disco y los monitores. La mayoría de las computadoras MS-DOS tienen conexiones (llamadas *puertas*) para comunicarse con otros dispositivos hardware. Los dos tipos más comunes de puertas son *serie* y *paralelo.* La diferencia se basa en los métodos internos que usa el dispositivo, que está conectado a la puerta, para comunicar con la computadora. Esta diferencia entre los dos métodos no afecta al

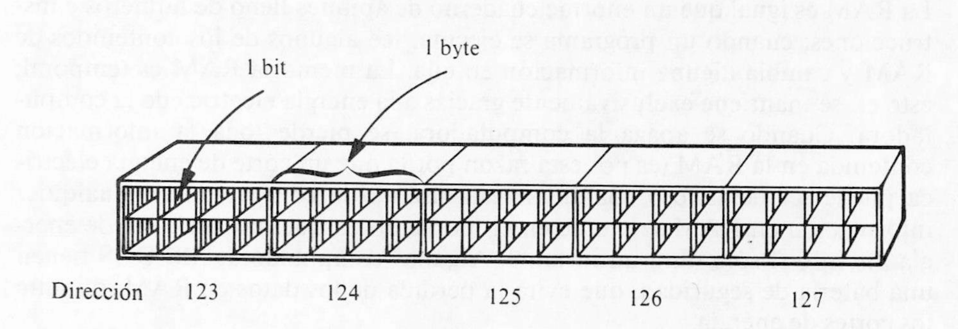


Figura 1-1. Ejemplo de bits y bytes

rendimiento del dispositivo, pero significa que hay que tener cuidado al seleccionar el tipo de dispositivo apropiado a las puertas de la computadora. No se puede conectar una impresora serie a una puerta de impresora de tipo paralelo o viceversa; no ocurriría nada. (Generalmente, sólo las impresoras se conectan en puertas de tipo paralelo; la mayor parte de los dispositivos externos restantes, tales como los modems, se conectan a puertas serie).

Cuando la CPU necesita comunicar con otro componente hardware, simplemente llama a la parte del sistema operativo que conoce el hardware. Todos estos procesos ocurren en cuestión de microsegundos —tan rápidamente que, a menudo, no se valora la importancia de lo que se está realizando—. Por ejemplo, si usted ejecuta un programa procesador de texto, el programa pedirá al MS-DOS cada carácter que usted pulse en el teclado. En este caso, el programa le dice al MS-DOS, «Estoy dispuesto para recibir un carácter; ¿ha pulsado uno el usuario? Si no, esperaré hasta que lo pulse».

Ejecución de programas de aplicación

Puede que se haya dado usted cuenta que el último ejemplo describía un proceso de comunicación no entre dispositivos hardware, sino entre el hardware y un software de aplicación. La segunda parte importante del papel del MS-DOS, como gestor del tráfico de información consiste en ayudar a que los programas se ejecuten sin problemas. Cualquier programa que usted ejecute puede pedir ayuda al sistema operativo en la comunicación con el hardware. Por ejemplo, un programa procesador de texto puede que necesite saber cuántas unidades de disco tiene usted en la computadora. En lugar de tener que conocer todas las órdenes de la computadora para solucionarlo, el programa puede preguntar al sistema operativo.

En su papel de intermediario entre la CPU y el programa de aplicación, el MS-DOS realiza dos tareas importantes: encuentra el programa que usted quiere ejecutar y lo coloca en RAM, y ayuda a que el programa realice I/O. Por ejemplo, cuando usted le dice al MS-DOS que quiere hacer procesamiento de textos, realmente le está ordenando que saque del disco el programa procesador de texto, lo cargue en RAM y le diga a la CPU dónde debe comenzar.

El sistema operativo también traduce una petición de un programa de aplicación a un lenguaje estándar, de forma que el mismo programa se pueda usar sobre diferentes computadoras. Esta es la razón por la cual usted puede usar algunos programas en computadoras construidas por distintos fabricantes. Por ejemplo, un programa puede decirle al sistema operativo: «Imprime estos caracteres en la pantalla», sin tener que saber nada de lo que requiere esa computadora particular para su visualización. Así, aunque las computadoras que ejecutan MS-DOS pueden ser completamente distintas, un programa se puede ejecutar sobre cualquiera de ellas, ya que el MS-DOS ayuda a la comunicación con el hardware de una forma estándar.

La mayoría de la gente que utiliza el MS-DOS como sistema operativo lo hace a causa de los muchos programas de aplicaciones que existen para él. Como se verá en el capítulo 6, hay miles de programas de aplicación que se ejecutan con el MS-DOS (esto es, que requieren el sistema operativo MS-DOS para poder ejecutarse). Sin embargo, ya que hay software escrito para otros sistemas operativos (tales como CP/M, que se trata más adelante en este capítulo), no todos los programas de aplicación se podrán ejecutar en su computadora. Usted tiene que estar seguro de que el programa que compra se ejecuta con el MS-DOS como sistema operativo, o, si no, usted no podrá ejecutarlo sobre su computadora MS-DOS. Puesto que algunos programas requieren cierto hardware, tal como una impresora específica o una cierta cantidad de RAM, usted debería estar seguro de que tiene el hardware adecuado antes de comprar cualquier software.

Hemos visto cómo el MS-DOS toma información de un dispositivo y la transfiere a otro. Este proceso ocurre tan comúnmente en la transferencia diaria de información entre ficheros, que merece la pena estudiar más a fondo lo que son los ficheros y cómo se almacenan en los discos.

COMO USA EL MS-DOS LOS DISCOS Y LOS FICHEROS

Cuando se guardan datos en un disco, la información se almacena en un fichero. Almacenar información en ficheros es lo mismo que organizar los datos en carpetas en un archivador. Un *fichero* es una colección de informaciones identificada con un nombre único que usted le asigna. Los ficheros son básicos para usar la computadora; sin ellos su trabajo se perdería cuando la computadora se apagara.

La información contenida en un fichero puede consistir en texto (como por ejemplo, una memoria o un informe), datos (tales como una lista de direcciones para envíos por correo) o un programa (como por ejemplo, un procesador de texto). El fichero puede tener cualquier longitud, limitada solamente por el espacio disponible en el disco en el que está almacenado. Cuando usted quiere que un programa opere sobre un fichero (bien para obtener información o bien para añadir información al fichero), simplemente usa la orden de acceso a un fichero y le dice al programa el nombre del fichero.

Los ficheros se almacenan en discos o en disquetes. Un *disco* es una pieza redonda de material rígido recubierto de un medio magnético; un *disquete* (o disco flexible) es una versión flexible de un disco. La mayoría de las discusiones de este libro se refieren a los disquetes, pero casi todas las órdenes del MS-DOS operan de la misma manera sobre discos que sobre disquetes. Este libro usará el término disquete para referirse específicamente a disquetes flexibles y el término disco para referencias más amplias que se aplican tanto a discos como a disquetes.

¿Qué ocurre cuando el MS-DOS recupera para usted los ficheros almacenados en un disco? El disco gira a alta velocidad mientras la cabeza se mueve hacia dentro y hacia fuera (la cabeza de un disco es similar a la cabeza de un magnetofón). El movimiento es similar a la selección de una canción en un tocadiscos. Cuando usted pide examinar un fichero, la cabeza va en primer lugar al *directorio,* un área especial del disco que contiene información acerca de cada fichero. Encuentra la posición del fichero que usted quiere y, a continuación, se mueve hasta el fichero.

Usted puede preguntarse cómo la cabeza del disco encuentra el fichero. En el directorio hay entradas que contienen dos números para cada fichero: la *pista* y el *sector.* El MS-DOS utiliza esta información para indicar con toda precisión la posición de cada fichero en el disco. El número de bytes en cada sector es constante en un disco, pero distintas unidades de disco pueden tener diferente número de bytes por sector y diferente número de sectores por pista. Afortunadamente, el MS-DOS hace que todo esto sea sencillo para usted.

La figura 1-2 ilustra las pistas y los sectores en un disquete. El número de

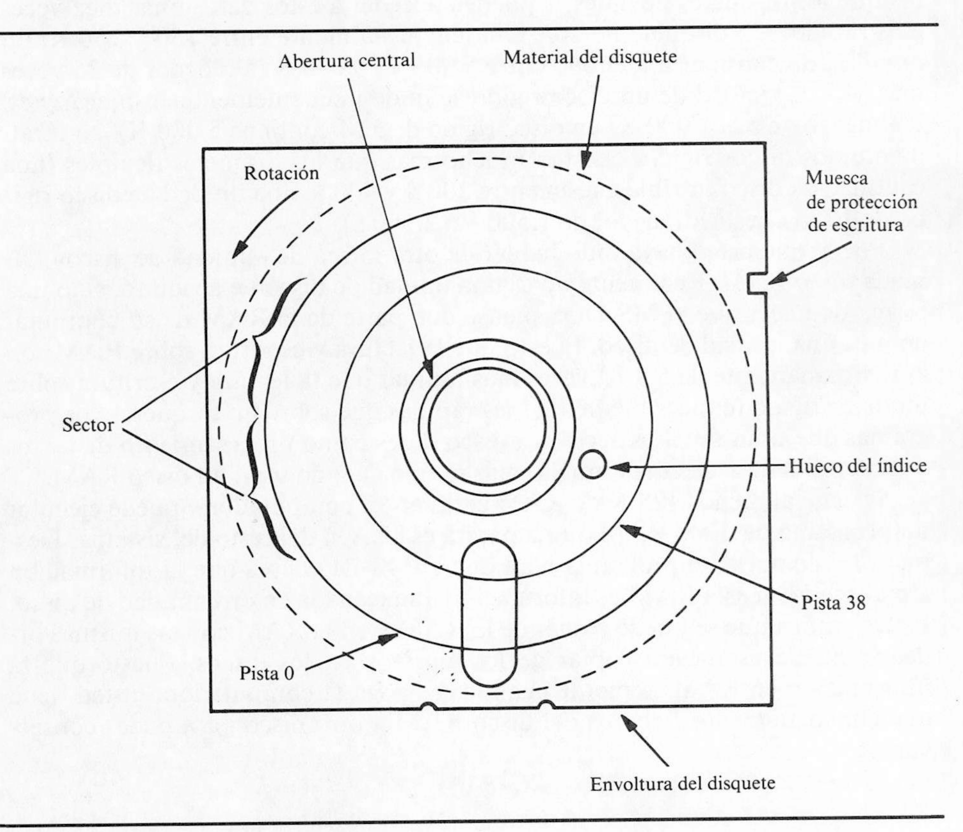


Figura 1-2. Pistas y sectores en un disco

pista es esencialmente una medida de la distancia del fichero al eje del disco. Las pistas son anillos concéntricos sobre el disco. Debido a que cada pista puede contener gran cantidad de información, las pistas se dividen en sectores. Continuando con la analogía con los discos de música, las pistas son como los surcos, y los sectores son como trozos de un surco. Los sectores se miden desde un punto fijo, en sentido contrario a las agujas del reloj, dando toda la vuelta y volviendo al punto de partida.

De esta manera, la entrada del directorio indica a la cabeza la distancia a la que tiene que buscar y el tiempo que debe esperar durante una rotación antes de comenzar a leer información. Encontrar una pista particular es similar a seleccionar una canción en un disco. Seleccionar el sector es lo mismo que buscar uno o dos compases de música dentro de una pista.

La *unidad de disco* es el mecanismo que contiene al disco. Casi todas las computadoras MS-DOS tienen al menos una unidad de disco. Los dos tipos principales de unidades de disco se diferencian por los tipos de discos que usan: discos flexibles o rígidos. Hay dos diferencias fundamentales entre los discos flexibles y rígidos: los discos rígidos o duros contienen muchos más datos que los disquetes flexibles, y pueden acceder a estos datos unas diez veces más rápido. Un disquete flexible contiene usualmente entre 150 y 750 K; un disco rígido contiene a menudo entre 5.000 y 20.000 K, alrededor de 25 veces más. (La capacidad de un disco rígido se mide frecuentemente en *megabytes*. Un megabyte son 1.000 K; un disco rígido de 5 M contiene 5.000 K). Naturalmente, los discos rígidos cuestan mucho más que los disquetes flexibles (una cantidad de disco flexible cuesta entre 200 $ y 400 $; una unidad de disco rígido de 10 M cuesta alrededor de 1.500 $ a 2.500 $).

Puede que usted haya oído hablar de otra forma de «unidad de disco» llamada *disco RAM*. Realmente no es una unidad de disco en absoluto, sino una forma de hacer que el MS-DOS piense que parte de la RAM de su computadora es una unidad de disco. Puesto que la lectura y escritura sobre RAM son aproximadamente de 5 a 10 veces más rápidas que la lectura y escritura sobre un disco rígido (e increíblemente más rápidas que sobre un disquete), los programas que usan muchos accesos a disco (tales como procesamiento de textos y contabilidad) se ejecutan mucho más rápido cuando usan un disco RAM.

Si tiene al menos 128 K de RAM extra en su computadora, puede ejecutar un programa de disco RAM que separará esa RAM del resto del sistema. Después de separarla, el programa hará que el MS-DOS crea que la información almacenada en esta RAM es información almacenada en otra unidad de disco. Esto significa que se puede copiar de la RAM y en la RAM con las mismas órdenes que se usan para copiar de los discos y en los discos. Puesto que la información en RAM se pierde cuando se apaga la computadora, usted tiene que copiar todos los ficheros del disco RAM a otro disco para poder conservarlos.

EL MS-DOS POR DENTRO

Mucha gente ejecuta el MS-DOS durante años sin saber nada acerca de lo que el MS-DOS está haciendo por ellos. Sin embargo, un ligero conocimiento de cómo trabaja el MS-DOS puede ayudarle a usar su sistema operativo con efectividad. También le puede ayudar a determinar los límites de lo que se puede esperar del MS-DOS.

Si usted pudiera mirar dentro del MS-DOS, probablemente vería una masa muy complicada de instrucciones. Estas instrucciones están escritas en *lenguaje máquina,* que es un lenguaje especial que la CPU sabe cómo leer (muchos programas de aplicación están escritos también en lenguaje máquina). Afortunadamente, usted no necesita conocer el lenguaje máquina para usar el MS-DOS. Ni necesita saber cómo hace el MS-DOS su trabajo. Sin embargo, el proceso por el cual el MS-DOS ejecuta programas no es difícil de entender y es útil saber algo de ello, especialmente cuando se dan órdenes directamente al MS-DOS.

Cómo ejecuta las órdenes el MS-DOS

El proceso que se trata en esta sección está relacionado con las funciones de utilidad del MS-DOS, y usted aprenderá a usar esas funciones en los próximos cinco capítulos. El MS-DOS es como un programa que está siempre funcionando. Cuando usted enciende la computadora, el MS-DOS pasa del disco a la RAM y comienza a ejecutarse (una descripción completa de esto aparece en el capítulo 2). Cuando el MS-DOS está dispuesto para que usted le dé una orden o ejecute un programa, visualiza una *petición de orden* (prompt) en la pantalla y espera que se le diga lo que debe hacer. Una petición de orden es simplemente una señal que indica que un programa (en este caso, el MS-DOS) está esperando a que se teclee algo.

La petición de orden del MS-DOS, usualmente A> o B>, indica que el *intérprete de órdenes* del MS-DOS está esperando que usted le diga qué hacer a continuación. El trabajo del intérprete de órdenes es leer las órdenes que se le dan al MS-DOS, encontrar el programa u orden que se quiere ejecutar y comenzar a ejecutarlo. (En el resto de la sección, usaremos la palabra «orden» aunque el proceso corresponda igualmente a cualquier programa que el MS-DOS ejecute).

Para ejecutar una orden, simplemente se escribe su nombre en el teclado. El MS-DOS visualiza los caracteres en la pantalla cuando usted los pulsa. A continuación, se pulsa la tecla RETURN, que normalmente está en la fila del medio de la columna derecha de las teclas alfabéticas. También puede estar marcada con un símbolo tal como una flecha que apunta hacia abajo y a la izquierda. En algunas computadoras, esta tecla puede tener el rótulo ENTER.

Después de que se le da al MS-DOS el nombre de la orden, el sistema operativo tiene que encontrar el programa correspondiente a la orden. Tiene dos

posibilidades para encontrarlo. Una orden puede estar en memoria interna o externa. Las *órdenes internas* están incorporadas en el propio MS-DOS. Esto significa que el MS-DOS no tiene que buscarlas en disco, puesto que están cargadas en RAM con el resto del MS-DOS. Otras órdenes son *externas*. Cada vez que se ejecutan estas órdenes, el MS-DOS tiene que leerlas del disco antes de poder ejecutarlas. (Alrededor de la mitad de las órdenes del MS-DOS son externas). Un *fichero para tratamiento por lotes o fichero «batch»* es un tipo especial de orden externa compuesta por un conjunto de órdenes del MS-DOS (tratado más adelante en el capítulo 2). La figura 1-3 muestra los pasos que sigue el sistema operativo cuando se le pide que ejecute una orden.

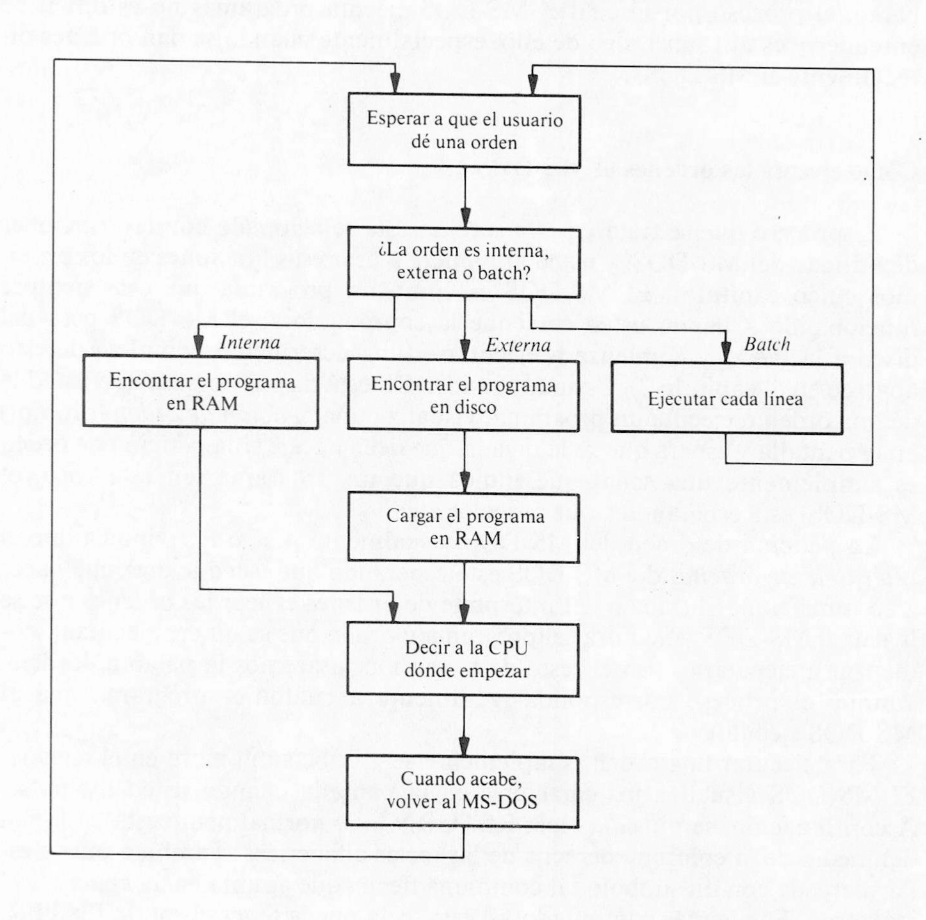


Figura 1.3. Cómo ejecuta el MS-DOS un programa

No hay diferencia en la forma en la que se le dice al MS-DOS que ejecute órdenes internas o externas. Por esta razón, frecuentemente usted no sabe qué tipo de orden le está pidiendo al MS-DOS que ejecute. Por ahora se pueden ignorar las diferencias, pero más tarde se verá por qué se necesita saber si una orden es externa o interna.

Cuando se ejecuta una orden externa, hay que estar seguro de que el disco que contiene esa orden está en la unidad de disco. Si no está en la unidad (o si se ha escrito mal el nombre de la orden), el MS-DOS dará el *mensaje de error* «Bad command or file name» (error en la orden o en el nombre del fichero). Los mensajes de error, como puede suponer por su nombre, son simplemente comunicaciones por las cuales el MS-DOS le hace saber que hay algo erróneo. El capítulo 5 trata de cómo evitar los mensajes de error y cómo interpretarlos.

Recuerde que el MS-DOS está siempre en RAM, aunque se le pida que ejecute una orden o un programa de aplicación. Cuando se ejecuta una orden, el MS-DOS la carga en la memoria adyacente al sistema operativo, como se ve en la figura 1-4. Si la orden quiere hacer algo que la CPU no puede manejar ella misma (tal como imprimir un carácter en la pantalla), llama a un subprograma, que es como una pequeña parte del sistema operativo, para que realice esa función. Vea la figura 1-5. En esta ilustración cada caja en el sistema operativo representa un subprograma. Cada caja en el programa de aplicación representa un paso del programa. Cuando se termina la orden, el sistema operativo permanece en RAM dispuesto para el próximo conjunto de instrucciones.

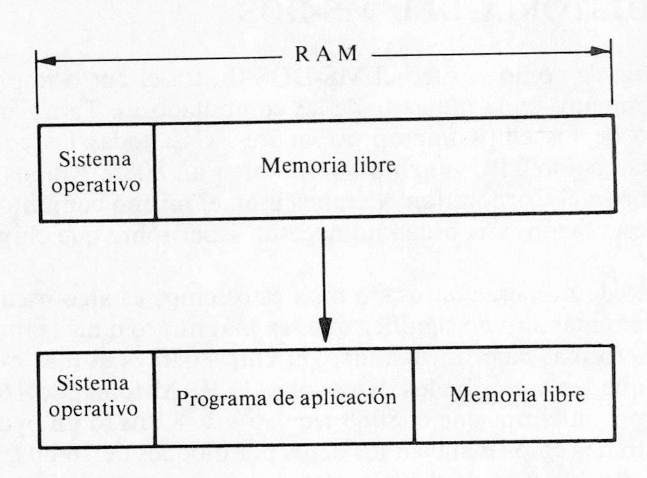


Figura 1-4. El sistema operativo cargando un programa en memoria

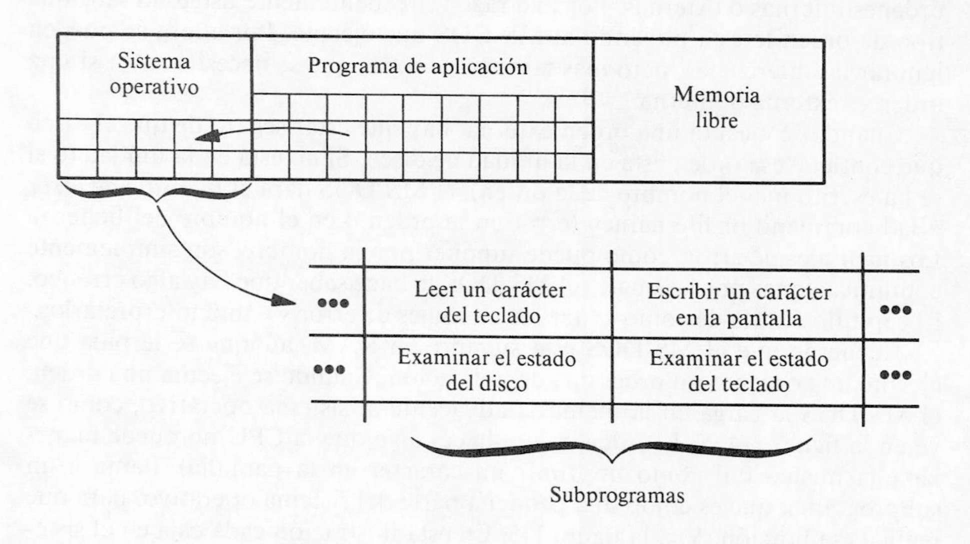


Figura 1-5. Llamada a un subprograma del sistema operativo

BREVE HISTORIA DEL MS-DOS

La historia de cómo se creó el MS-DOS ilustra el curso impredecible de los acontecimientos en la industria de las computadoras. También está unida al desarrollo de los chips microprocesadores. Casi todas las computadoras MS-DOS usan como CPU un chip 8086 o bien un 8088. Ambos chips están diseñados por Intel Corporation, y ambos usan el mismo conjunto de instrucciones. Por esta razón, una orden no necesita saber sobre qué chip de CPU se va a ejecutar.

El sistema de numeración usado para estos chips es algo oscuro: un chip con un número más alto no significa que sea más nuevo o más rápido que otro con un número más bajo. En realidad, el chip 8086 es el más rápido de los dos, puesto que lee y escribe los datos sobre la RAM tomando 16 bits (o dos bytes) cada vez, mientras que el 8088 lee datos de 8 bits (o un byte) cada vez. Puesto que ambos chips manejan los datos por bloques de 16 bits, el 8088 tiene que leer dos bloques de 8 bits antes de poder procesar la misma información.

El procesador 8086 es importante en la historia del MS-DOS, ya que el MS-DOS fue diseñado originalmente por Tim Paterson y Seattle Computer

Products en 1980 para ser el sistema operativo de su tarjeta de CPU, recién diseñada, que llevaba un procesador 8086. Cuando la tarjeta de CPU de Seattle Computer apareció en el mercado, hacia la mitad de 1979, el MS-DOS no estaba ni siquiera en los tableros de diseño. Digital Research había anunciado que el sistema operativo CP/M-86 estaría dispuesto pronto para ejecutarse sobre el sistema 8086 y así las expectativas eran que no se necesitaría ningún otro sistema operativo. (El sistema operativo CP/M de Digital Research era en ese tiempo el sistema operativo más popular para las computadoras que usaban el microprocesador 8080 o el Z80).

Sin embargo, la llegada del CP/M-86 se retrasó y, después de esperar casi un año, Seattle Computer decidió escribir su propio sistema operativo llamado QDOS. Cuatro meses más tarde, en agosto de 1980, el QDOS estaba disponible para los usuarios del Seattle. Poco tiempo después de su lanzamiento, otra empresa basada en Seattle, llamada Microsoft, decidió comprar el QDOS y comercializarlo como su propio sistema operativo, bajo el nombre de MS-DOS. Microsoft se había hecho famoso por su versión de BASIC, pero nunca había vendido un sistema operativo. Pocos meses después del lanzamiento del MS-DOS apareció el CP/M-86.

Microsoft ha lanzado nuevas versiones del MS-DOS cada vez que hacía mejoras en el programa. Cada lanzamiento posterior del MS-DOS se denomina nueva versión; estas versiones están numeradas, tal como «versión 1» y «versión 2». La primera versión del MS-DOS se llamó 1.0; cuando se hicieron mejoras, Microsoft lanzó las versiones 1.1 y 1.2. Estas revisiones, con numeración más alta, no contenían ni órdenes ni características nuevas. Sin embargo, la versión 2.0 lanzada al principio de 1983, tenía muchas características nuevas. La versión 2.1, lanzada al final de 1983, arregló los pocos fallos de la versión 2.0. Este libro usará números enteros (1 y 2) para referirse a las principales versiones del MS-DOS.

Puede que usted esté familiarizado con otros sistemas operativos, tales como el CP/M-80. Si usted ha utilizado CP/M, puede que se pregunte qué es lo que el MS-DOS tiene en común con él. Hay muchas similaridades ya que el CP/M era el sistema operativo más popular en el momento en que se diseñó el MS-DOS. La mayoría de los conceptos básicos son los mismos que los del CP/M, o bien simples extensiones de éstos. Por ejemplo, ambos sistemas operativos toman las órdenes de la misma manera: el sistema operativo hace una petición de orden (realmente, ambos usan A>), y el usuario escribe una orden. La estructura de la orden es la misma: el nombre de la orden seguido de instrucciones más específicas llamadas *argumentos*. Ambos sistemas operativos tienen órdenes internas y externas.

Los argumentos de las órdenes del MS-DOS usualmente son mucho más fáciles de recordar y usar que los de CP/M. Los nombres de las órdenes MS-DOS son también más fáciles de recordar. Por ejemplo, la orden MS-DOS para copiar ficheros se llama COPY; en CP/M, se llama PIP. Muchas de las órdenes MS-DOS proporcionan más información que las órdenes similares del CP/M. El MS-DOS tiene otras características útiles que le faltan

al CP/M. Si usted está acostumbrado a usar CP/M, probablemente le llevará muy poco tiempo aprender el MS-DOS.

LO QUE EL MS-DOS NO PUEDE HACER

Naturalmente, el MS-DOS no es el sistema operativo «definitivo», puesto que no incluye todas las posibles características que un sistema operativo puede tener. Cuando Microsoft mejora el MS-DOS le añade nuevas características. En la discusión previa sobre los sistemas operativos se vio lo que puede hacer el esqueleto básico de un sistema operativo: ayudar a la CPU a interactuar con el hardware y ejecutar programas. Sin embargo, algunos otros sistemas operativos pueden hacer mucho más que eso. Consideremos las dos siguientes situaciones.

¿Qué hacer si tiene un trabajo que requiere usar dos programas separados? Por ejemplo, puede querer incluir algunos datos contables en un informe o memoria que está escribiendo. Con las versiones 1 y 2 del MS-DOS, sólo se puede ejecutar un programa cada vez: usted debería ejecutar primero su programa de contabilidad para obtener los resultados y, a continuación, ejecutar el procesador de texto para escribir la memoria. Futuras versiones del MS-DOS pueden permitir que ambos programas se ejecuten al mismo tiempo. Otros sistemas operativos sofisticados también permiten esta facilidad. La capacidad para ejecutar varios programas simultáneamente se denomina *multitarea* o *concurrencia.*

Veamos la segunda situación. ¿Qué hacer si usted trabaja en una oficina y hay mucha gente que quiere usar una única computadora? El MS-DOS es un sistema operativo *monousuario.* Por esta razón, cuando ejecuta el MS-DOS, usted es la única persona que puede darle órdenes. Un sistema operativo *multiusuario* permite que muchos usuarios en terminales diferentes puedan ejecutar programas al mismo tiempo, compartiendo así todos los recursos de la computadora. (Algunos sistemas operativos, tales como UNIX, son a la vez multiusuario y multitarea). Puesto que Microsoft lanza regularmente nuevas versiones del MS-DOS, una versión multiusuario puede estar pronto disponible. Realmente, Microsoft ya vende XENIX, su versión de UNIX, y ha dicho que los dos sistemas operativos se fusionarán en el futuro.

Ahora que sabe lo que es el MS-DOS y por qué el MS-DOS es importante en su computadora, usted está preparado para aprender a usarlo. Naturalmente, usar el MS-DOS es mucho más interesante que aprenderlo. El próximo capítulo le permitirá aprenderlo y le presentará la información básica necesaria para empezar a usar el MS-DOS.

2

MANEJO DEL MS-DOS: INTRODUCCION A LAS ORDENES Y LOS FICHEROS

Puesta en marcha del MS-DOS
Qué hacer cuando vea la petición de orden A >
Utilización de argumentos en las órdenes
Algunas órdenes sencillas: FORMAT y COPY
Utilización de ficheros en órdenes y programas
Utilización de ficheros batch para combinar órdenes del MS-DOS
Una sesión de ejemplo con el MS-DOS

La mayoría de la gente piensa que el MS-DOS es bastante fácil de aprender, una vez que han entendido unos cuantos conceptos. Este capítulo le explicará cómo

- Poner en marcha su computadora.
- Usar las órdenes del MS-DOS y ejecutar programas de aplicación.
- Usar ficheros.
- Construir ficheros batch.

Además, este capítulo introduce varias órdenes del MS-DOS y muestra cómo usarlas para realizar algunas tareas comunes. No intenta presentar todas las órdenes del MS-DOS, ni siquiera todas las formas diferentes de usar las órdenes que trata. Para obtener esas informaciones véase el capítulo 3, «Ordenes del MS-DOS». El capítulo 3 trata todas las órdenes en profundidad, incluyendo tanto sus usos habituales como los no habituales. Puede usarlo como referencia para consultar acerca de cualquier orden.

La mayoría de las órdenes que se introducen en este capítulo se acompañan con ejemplos. Pruebe las órdenes de los ejemplos en su computadora. Se presentan primero las órdenes más elementales. Puesto que la mayoría de las órdenes del MS-DOS tienen nombres muy relacionados con las acciones que realizan, el uso de cada orden suele ser fácil de recordar. Si necesita referirse a ellas más tarde, también están explicadas con detalle en el capítulo 3.

PUESTA EN MARCHA DEL MS-DOS

No hay un método estándar para cargar el MS-DOS en una computadora cada marca de computadora tiene su propio método. Normalmente el sistema operativo se carga insertando el disco del sistema operativo en una de las unidades de disco, antes de encender la computadora. Si su sistema hace la *carga inicial* (boot) del MS-DOS desde un disquete, usted usa generalmente el mismo disquete (llamado *disco del sistema*) cada vez que pone en marcha el sistema. La derivación histórica del término «boot» viene del refrán «pulling yourself up by your own bootstraps» que significa «hacerse a sí mismo». El tiempo que tarda la carga inicial varía de unas computadoras a otras.

Algunas computadoras usan métodos de puesta en marcha diferentes para cargar el MS-DOS en memoria. Por ejemplo, muchas computadoras que tienen discos duros leen el sistema operativo del disco duro, y no requieren el disco del sistema. Si tiene un sistema con disco duro y quiere hacer la carga inicial del MS-DOS directamente, puede que necesite copiar el sistema operativo sobre el disco duro. Debería consultar las instrucciones exactas en el manual que viene con su computadora.

Después de cargar el MS-DOS, probablemente la primera cosa que verá es una petición «Enter new date»: (Introduzca la nueva fecha). Esto ocurrirá a no ser que su disco contenga un fichero especial llamado *AUTOEXEC.BAT*, que consiste en una serie de órdenes escritas en un fichero especial que se ejecuta automáticamente, cuando se hace la carga inicial del sistema. Los ficheros AUTOEXEC.BAT se tratan en profundidad al final de este capítulo.

La petición de fecha tiene un aspecto similar a este:

```
Current date is Tue 1-01-1980
Enter new date:
```

El bloque sombreado que aparece a la derecha de la petición de fecha se llama *cursor*. El cursor es un bloque o un subrayado en la pantalla que muestra dónde aparecerá el siguiente carácter que usted pulse. Observe que el cursor está a continuación de los dos puntos en la segunda línea.

Escriba la fecha como tres números separados por guiones (-) o barras (/). A continuación, se muestran los formatos válidos para el 14 de septiembre de 1984. Observe que tiene que poner lo primero el mes, después el día y después el año.

```
9-14-84
9/14/84
09/14/84
```

Si se equivoca al escribir, pulse la tecla BACKSPACE (a veces rotulada con una flecha apuntando a la izquierda ←) para volver sobre los caracteres incorrectos. Cuando termine pulse la tecla RETURN.

¿Por qué el MS-DOS quiere conocer la fecha? Cuando el MS-DOS quiere conservar ficheros en el disco, graba tanto la fecha como la hora, de forma que usted pueda saber cuándo creó el fichero. Sin embargo, no es obligatorio grabar la fecha y la hora. Si no quiere molestarse en introducir la fecha correcta, simplemente pulse RETURN y el MS-DOS asumirá que la fecha es el 1 de enero de 1980 (o cualquier otra que haya sido visualizada en la pantalla). A continuación el MS-DOS le pide la hora.

```
Current time is 0:00:12.83
Enter new time:
```

Como en la petición de fecha, el MS-DOS espera que usted responda utilizando un formato específico. El único formato aceptable es en horas, minutos y segundos, separados por dos puntos (:). Las horas se guardan en un formato de 24 horas. Por ejemplo, las 2 de la tarde son las 14, y la hora antes de medianoche son las 23.

Puesto que usted raramente necesita una hora muy exacta, no hace falta que incluya los segundos. Si los incluye, puede introducir también centésimas de segundo, poniendo un punto a continuación de los segundos y después las centésimas de segundo. A continuación se muestran algunos ejemplos de horas válidas:

```
10:11:12
15:45
0:0:0.5
```

El primer ejemplo indica 12 segundos después de las 10:11 de la mañana; el segundo indica las 3:45 de la tarde; y el tercero corresponde a medio segundo después de medianoche.

Se usa BACKSPACE para corregir equivocaciones y, si no se quiere grabar la hora correcta, se pulsa RETURN sin haber escrito nada. No se preocupe si ha introducido una fecha u hora equivocadas; hay órdenes (llamadas apropiadamente DATE y TIME) para cambiarlas fácilmente.

Después de introducir la fecha y la hora, verá un mensaje de inicialización que identifica la versión del DOS. La petición de orden A> aparecerá a continuación del mensaje de inicialización.

```
The IBM Personal Computer DOS
Version 2.00 (C)Copyright IBM Corp 1981, 1982, 1983

A>
```

En la petición de orden, algunas computadoras utilizan después de la A dos puntos (:) en lugar del signo de mayor que (>).

La petición de orden estándar, A>, es la que verá después de darle al MS-DOS la fecha y la hora. A> es la forma en que el MS-DOS le dice que está esperando a que usted le dé una orden. Cuando vea A>, sabrá que no se está ejecutando ningún otro programa.

QUE HACER CUANDO VEA LA PETICION DE ORDEN A>

Cuando la petición de orden A> esté en su pantalla, el siguiente paso consistirá en darle una orden al MS-DOS. En general, para dar una orden o ejecutar un programa, se escribe el nombre de la orden y se pulsa la tecla RETURN. (Hay otras cosas que se pueden escribir después del nombre de una orden; esto se tratará más adelante en este capítulo). La primera orden que quizá quiera usted dar es la orden DIR, que visualiza una lista de todos los ficheros que hay en el disco. El siguiente ejemplo le muestra cómo hacerlo. (En este libro los ejemplos están separados del resto del texto mediante corchetes. Se usarán a lo largo del libro para ilustrar las órdenes o procedimientos tratados en el texto.

Cuando vea la petición de orden A>, escriba DIR. (Puede escribirlo tanto en mayúsculas como en minúsculas). Su entrada aparecería en la pantalla en una forma similar a esta:

```
A>DIR
```

Observe que lo que usted ha escrito está subrayado en el ejemplo. En todo este libro, lo que usted escribe aparece subrayado para distinguirlo fácilmente de los mensajes generados por el MS-DOS.

A continuación, pulse RETURN. La orden DIR se ejecuta y escribe una lista de todos los ficheros contenidos en la undidad de disco A:. Usted acaba de dar su primera orden al MS-DOS.

La orden DIR es una orden interna. Recuerde que, como se vio en el capítulo 1, las órdenes internas están siempre en memoria y que el MS-DOS no tiene que buscarlas en los discos.

Cuando se escriben las órdenes, se pueden usar algunas teclas especiales para recibir ayuda. Se puede pulsar BACKSPACE para borrar un carácter equivocado (como se hacía en la petición de fecha y hora). Si quiere eliminar toda una línea, pulse la tecla rotulada con ESC o ESCAPE.

Sí, cuando intenta ejecutar una orden, el MS-DOS le da un mensaje de error tal como «Bad command or file name», que significa «orden o nombre de fichero erróneos», no se preocupe. Esto indica que se ha producido el error más común: simplemente se ha escrito mal el nombre del programa. Algunos de los mensajes de error más comunes están tratados en el capítulo 5.

La unidad por defecto y las órdenes externas

Como hemos visto, la petición de orden del MS-DOS indica que el sistema operativo está esperando una orden. La petición de orden también indica algo más. Indica en qué unidad de disco buscará automáticamente el MS-DOS las órdenes y los ficheros. Esta unidad se denomina *unidad por defecto.* «Por defecto» significa algo que se asume a no ser que se especifique otra cosa.

Cuando el MS-DOS hace una petición de orden con A>, la unidad por defecto es la unidad A:. Cuando la hace con B> o con C>, indica que la unidad por defecto es la B: o la C: (las letras correspondientes a los discos están seguidas usualmente por dos puntos para diferenciarlas de los nombres de ficheros).

Puede que usted se pregunte de dónde vienen las letras A: y B:. Su computadora puede tener muchas unidades de disco o bien sólo una. La disposición de las unidades de disco se hace usualmente de arriba a abajo o de izquierda a derecha. A los discos se les asignan letras comenzando con la A:, que es el primer disco de su computadora; B: es el segundo y así sucesivamente. El manual de su computadora debería explicar cómo se asignan las letras a los discos. Si tiene un disco duro, el MS-DOS puede tratarlo como si estuviera dividido en discos más pequeños, cada uno con su propia létra. La figura 2-1 muestra algunas posibles organizaciones.

Cuando se da una orden interna como DIR, no es importante saber qué unidad contiene el disco del sistema MS-DOS. Sin embargo, es distinto cuando se da una orden externa, ya que el MS-DOS tiene que leer del disco las órdenes externas antes de poder ejecutarlas. El MS-DOS normalmente buscará en la unidad por fefecto cuando intente leer la orden.

Organización # 1:

Discos:	Flexible 1	Flexible 2		
Letras:	A:	B:		

Organización # 2

Discos:	Flexible 1	Disco duro 1		
Letras:	A:	C:		

Organización # 3

Discos:	Flexible 1	Flexible 2	Disco duro 1	Disco duro 2
Letras:	A:	B:	C:	D:

Organización # 4

Discos:	Flexible 1	Disco duro 1	Disco duro 2	Disco RAM
Letras:	A:	B:	C:	D:

Figura 2-1. Posibles métodos de asignación de letras a los discos.

Por ejemplo, cuando usted listó el directorio del disco del sistema, probablemente vio un fichero llamado BASIC.COM. La orden BASIC es una orden externa que permite escribir programas en BASIC. Para ejecutar una orden externa simplemente se escribe su nombre, igual que cuando se ejecuta una orden interna.

Por ejemplo, para comenzar a ejecutar el programa BASIC introduzca la orden

```
A>BASIC
```

(Recuerde que, cuando se dan órdenes al MS-DOS, hay que pulsar RETURN después de escribir la orden). Se puede oír el zumbido de la unidad de disco cuando lee BASIC. El MS-DOS carga el programa BASIC y comienza a ejecutarlo.

Para salir del programa BASIC, escriba la palabra SYSTEM y pulse RETURN. Entonces verá de nuevo la petición de orden A>.

Cuando se le dice al MS-DOS que ejecute una orden externa, éste buscará sólo en el disco de la unidad por defecto, a no ser que se le diga otra cosa. Si las órdenes que usted quiere ejecutar están en el disco B:, puede que quiera cambiar la unidad por defecto de A: a B:. Se puede cambiar la unidad por defecto dando otra letra seguida por dos puntos y pulsando RETURN.

Por ejemplo, para cambiar la unidad por defecto de A: a B:, escriba la orden

```
A>B:

B>
```

La petición de orden B> muestra que la unidad por defecto ha sido cambiada a B:.

Si el programa no está en la unidad por defecto y no se dijo en qué disco buscarlo, el MS-DOS escribirá en la pantalla «Bad command or file name».

Cambiar la unidad por defecto es una forma de ejecutar una orden de un disco distinto, pero hacer esto a menudo puede ser incómodo. Afortunadamente, hay una forma más fácil de decirle al MS-DOS dónde buscar algo. Simplemente escriba el nombre de la unidad de disco antes del programa.

Por ejemplo, si la unidad por defecto es B:, y BASIC.COM está en A:, introduzca la orden

```
B>A:BASIC
```

Esto le dice al MS-DOS que no busque en la unidad por defecto, sino en A:. Esto es más fácil que cambiar la unidad por defecto, que se haría como sigue:

```
B>A:

A>BASIC
```

UTILIZACION DE ARGUMENTOS EN LAS ORDENES

Las órdenes del MS-DOS que se han usado hasta ahora son verdaderamente simples. Es posible dar más información al MS-DOS en las órdenes, para que pueda realizar tareas más complicadas.

Para dar al MS-DOS más información en las órdenes se utilizan modificantes llamados *argumentos.* Casi todas las órdenes MS-DOS tienen argumentos obligatorios u opcionales. El saber cómo usar argumentos le permite usar su computadora con más efectividad.

Los argumentos se usan para modificar tanto las órdenes internas como las externas. Su función es muy similar a la de los modificantes en el lenguaje hablado. Por ejemplo, en vez de decirle a alguien «compra», usted puede decir

«compra un periódico en la tienda». El argumento no modifica el trabajo realizado por la orden, sino que da a la orden más información con la que trabajar.

De la misma forma, cuando usted da una orden al MS-DOS, puede también proporcionarle instrucciones específicas acerca de lo que quiere. Para hacer esto, se da primero el nombre de la orden y a continuación se añaden argumentos para modificar la orden. Es algo similar a la escritura de frases cortas en inglés, salvo que cada sentencia comienza con el verbo (como en «copy these files from this disk to that one» que significa «copia estos ficheros de este disco a este otro» o «print this file» que significa «imprime este fichero»).

Por ejemplo, usted ya vio cómo la orden DIR visualiza la lista de ficheros contenidos en el disco:

```
A>DIR
```

La orden DIR, sin argumentos, lista los ficheros de la unidad por defecto, que es la A: en este caso. Se puede dar a la orden un argumento opcional que define además de qué unidad se quiere ver el directorio. Por ejemplo:

```
A>DIR B:
```

Aquí B: es un argumento que instruye a la orden DIR para que visualice los ficheros de B: en vez de los de la unidad por defecto.

La orden DIR lleva un *argumento opcional* puesto que puede funcionar con o sin argumento. Algunas órdenes, sin embargo, llevan *argumentos obligatorios* porque necesitan saber sobre qué se pretende que trabajen.

Por ejemplo, la orden TYPE, que escribe el contenido de un fichero en la pantalla, necesita que se le diga el nombre del fichero que se quiere escribir. Suponga que quiere ver el texto de un fichero llamado ACCNTS.RPT. Debería dar la orden

```
A>TYPE ACCNTS.RPT
```

Los contenidos de ACCNTS.RPT se visualizarían en la pantalla. Si simplemente se introduce la orden TYPE sin argumento, el MS-DOS respondería con un mensaje de error «Invalid number of parameters» (número incorrecto de parámetros).

No todos los ficheros pueden ser visualizados en forma legible utilizando la orden TYPE. Si se intenta escribir un fichero que no está escrito en el conjunto de caracteres estándar, denominado American Standard Code for Information Interchange (ASCII), la computadora sonará y visualizará caracteres ilegibles en la pantalla.

Hay muchos tipos de argumentos que se pueden dar a las órdenes; los más comunes son nombres de unidades de disco o nombres de ficheros. En los dos ejemplos previos, la orden DIR lleva un nombre de unidad (B:) mientras que la orden TYPE lleva un nombre de fichero (ACCNTS.RPT). Los otros tipos de argumentos se explican en el capítulo 3.

ALGUNAS ORDENES SENCILLAS: FORMAT Y COPY

Ahora que sabe cómo poner en marcha el MS-DOS y darle órdenes, usted está preparado para utilizar la computadora en cosas más prácticas. La primera tarea que necesita aprender es cómo preparar y copiar los discos. En esta sección se usarán algunas órdenes más del MS-DOS.

Preparación de los discos con la orden FORMAT

La mayoría de los discos no están preparados para su uso cuando se compran. Necesitan primero ser *formateados.* El formateado consiste en poner marcas electrónicas en el disco, de forma que el MS-DOS sepa encontrar los lugares correctos donde colocar los datos. Tenga en cuenta, sin embargo, que todo lo que hubiera previamente en el disco se pierde con la operación de formateado.

Formatear un disco es similar a pintar líneas blancas en una zona de aparcamiento. Igual que las líneas indican dónde aparcar los coches, el modelo de formato en un disco le dice al MS-DOS dónde puede poner los datos. Si el aparcamiento es nuevo, no habrá líneas previas; sin embargo, si hay líneas viejas, lo primero que hay que hacer es pintar de negro toda la zona, para que las nuevas líneas queden claras. Al pintar de negro la zona o al formatear el disco, toda la «información» anterior se pierde.

Cuando se formatea un disco usado, toda la información que había en el disco se borra. Por esta razón, cuando use la orden FORMAT, tenga especial cuidado de no usar un disco que contenga información que usted pueda necesitar en el futuro. Compruebe el contenido de un disco usado con la orden DIR, antes de formatearlo. Los discos duros usualmente vienen ya formateados; es muy improbable que usted necesite alguna vez formatear un disco duro.

La orden FORMAT lleva como argumento el nombre de la unidad que se quiere formatear.

Por ejemplo, para formatear un disquete en la unidad B:, inserte un disquete nuevo, en blanco, y dé la orden

```
A>FORMAT B:
```

Esto formateará el disquete de la unidad B:.

En breve, veremos otro de los argumentos de la orden FORMAT.

Copia de ficheros con la orden COPY

La orden COPY permite copiar ficheros de un disco a otro. Lleva, al menos, dos argumentos: el nombre del fichero que se quiere copiar y el nombre de la unidad en la cual copiarlo. (Realmente, la orden COPY puede hacer mucho más, y sus otras funciones se describen en el capítulo 3).

Por ejemplo, si usted da la orden

```
A>COPY BASIC.COM B:
```

el MS-DOS copiará el fichero BASIC.COM del disco A: al disco B:

Utilización de FORMAT y COPY para hacer copias de seguridad del disquete del sistema

Ahora que sabe cómo usar las órdenes FORMAT y COPY puede aplicarlas. En esta sección, se hará una copia de seguridad del disquete del sistema MS-DOS (el disquete que se usa para la carga inicial del MS-DOS).

¿Por qué hacer una copia del disquete del sistema? Nunca es prudente confiar en un disquete único para las necesidades cotidianas. Demasiadas cosas pueden ocurrir. Un borrado accidental, una taza de café derramada, incluso un corte de corriente, mientras se está leyendo o grabando un disco, pueden dañar el disquete o bien los ficheros importantes que contiene.

Para evitar una catástrofe imprevista, debería hacer una copia de este disquete y a continuación usar la copia en vez del original. Si algo le ocurre a la copia, puede hacer otra copia del original. Las órdenes FORMAT y COPY permiten hacer copias del disquete del sistema MS-DOS.

Para copiar el disquete del sistema, lo primero que se necesita es formatear un disquete utilizando la orden FORMAT con un argumento especial llamado /S. El argumento /S hace que la orden FORMAT cree en el disquete un área llamada *pistas de carga inicial* y que coloque unos ficheros especiales llamados *ficheros del sistema* (también llamados *ficheros ocultos* porque no se listan cuando se da la orden DIR). Las pistas de carga inicial y los ficheros del sistema no se necesitan en los disquetes que se usan para guardar ficheros de datos, pero sí se necesitan en un disquete del sistema porque contienen partes esenciales del sistema operativo MS-DOS.

Una vez que el nuevo disquete ha sido formateado con el argumento /S, el siguiente paso consiste en copiar todos los ficheros del disquete del sistema al nuevo disquete.

Por ejemplo, si tiene una computadora con dos discos, primero ponga el disquete del sistema en la unidad A: y un disquete nuevo, en blanco, en la unidad B:. A continuación dé la orden

```
A>FORMAT B: /S

Formatting ... format complete
```

El disquete de la unidad B: es ahora utilizable en la carga inicial (esto es, se puede poner en marcha la computadora con él). Sin embargo, no contiene todavía los muchos ficheros del MS-DOS que aparecen en el listado del directorio. Para copiar estos ficheros, dé la orden

```
A>COPY A:*.* B:
```

(El *.* utilizado en la orden COPY se describe más adelante. Indica al MS-DOS que copie todos los ficheros de A:, sin importar sus nombres).

Otra forma de hacer una copia completa del disquete del sistema es con la orden DISKCOPY. Esta orden también se describe en el capítulo 3.

Copia de seguridad parcial del disquete del sistema

Probablemente, no todas las órdenes incluidas en el disquete del sistema MS-DOS le serán necesarias en su trabajo cotidiano. Si quiere hacer una segunda copia de seguridad solamente con estas órdenes, tendrá más espacio para otros ficheros.

Para determinar qué órdenes quiere tener en el disco, consulte la tabla 2-1. Esta tabla lista las órdenes externas del MS-DOS, da su tamaño aproximado

Tabla 2-1. Ordenes externas del MS-DOS

Orden	Tamaño (en K)	De uso frecuente	Específicas de IBM
ASSIGN	1		
BACKUP	4	*	*
BASICA	26	*	
CHKDSK	6	*	
COMP	3		
DEBUG	12		
DISKCOMP	3		*
DISKCOPY	2	*	*
EDLIN	5		
EXE2BIN	2		
FDISK	6		*
FIND	6		
FORMAT	6	*	
GRAPHICS	1		*
LINK	40		
MODE	3	*	*
MORE	1		
PRINT	5	*	
RECOVER	2		
RESTORE	4	*	*
SORT	1		
SYS	1		
TREE	2		*

en el disco (en K), e indica si las órdenes son suficientemente frecuentes como para ponerlas en el disco del sistema. Se ha listado el tamaño para que usted pueda determinar qué porcentaje del disco del sistema será ocupado por estos ficheros. La tabla 2-1 también indica qué órdenes son específicas de IBM. (Otros fabricantes suelen tener las mismas órdenes, pero a veces con otros nombres). Puesto que la mayor parte de las órdenes externas de uso frecuente son muy cortas, se pueden colocar todas ellas en un disquete con algún espacio de sobra. Si usted tiene muchos programas de aplicación diferentes, puede poner cada uno en un disquete distinto.

Para hacer una copia de seguridad parcial del disquete del sistema, simplemente siga las instrucciones del FORMAT /S y a continuación copie las órdenes de uso frecuente que aparecen en la tabla.

Protección de escritura en el disquete del sistema

Ahora que tiene una copia de seguridad del disquete del sistema MS-DOS, debería protegerla contra un borrado accidental o contra modificaciones involuntarias. Puede hacer esto con una *protección de escritura.*

Si su computadora usa disquetes de 5 1/4 pulgadas, quizá se haya dado cuenta de la muesca que hay en la parte derecha del disquete, aproximadamente a unos dos centímetros y medio de la parte de arriba. Esta es la *muesca de protección de escritura* (mostrada en la figura 2-2). Cuando se cubre esta

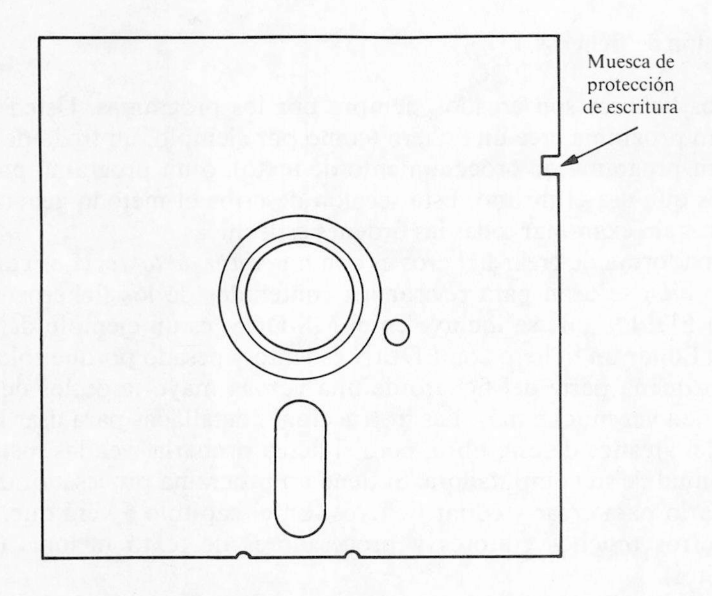


Figura 2-2. Muesca de protección de escritura en un disquete.

muesca con cinta adhesiva o con etiquetas autoadhesivas (normalmente incluidas en las cajas de disquetes), el MS-DOS protege el disquete no permitiendo que se escriba sobre él. Sin embargo, el MS-DOS puede aún leer el disquete.

Es una buena costumbre proteger los disquetes sobre los que no se quiere escribir: tanto los discos que vienen con la computadora como las copias del disquete del sistema. Naturalmente, usted puede despegar la etiqueta y escribir información sobre el disco si lo necesita. La idea básica que hay tras la protección de escritura de un disquete es reducir la posibilidad de destruir accidentalmente información importante.

UTILIZACION DE FICHEROS EN ORDENES Y PROGRAMAS

Probablemente, el uso más frecuente de las órdenes del MS-DOS será el de ayudarle a crear, mantener y mover ficheros en la computadora. Naturalmente, hay mucho más que saber acerca de las órdenes. Sin embargo, para usar eficazmente el MS-DOS, sólo se necesita conocer cómo usar unas cuantas órdenes de uso frecuente. La siguiente discusión presenta brevemente algunas de ellas. Puesto que esta sección trata de ficheros y no de órdenes, no se presentan todos los usos de dichas órdenes.

Creación de ficheros

Los ficheros son creados siempre por los programas. Usted puede hacer que un programa cree un fichero (como por ejemplo, un trozo de texto creado por un programa de procesamiento de texto), o un programa puede crear ficheros que usa él mismo. Esta sección describe el método general para crear ficheros sin examinar todas las órdenes específicas.

Una forma de crear ficheros es con un *editor de texto.* (Los editores de texto también se usan para revisar los contenidos de los ficheros de texto). La orden EDLIN, que se incluye en el MS-DOS, es un ejemplo de un editor de texto. Editar un fichero con EDLIN es lento y pesado porque sólo permite ver una pequeña parte del fichero de una vez; la mayoría de los demás editores permiten ver mucho más. Las instrucciones detalladas para usar EDLIN escapan del alcance de este libro, pero si desea probarla, vea las instrucciones en el manual de su computadora. Si tiene un programa procesador de texto, puede usarlo para crear y editar ficheros. En el capítulo 6 verá que hay disponibles otros muchos editores y procesadores de texto mejores (algunos son gratuitos).

La mayor parte de sus programas de aplicación también crearán ficheros. Por ejemplo, un programa de contabilidad creará ficheros para guardar sus

cantidades, y un programa de hoja de cálculo creará ficheros para guardar sus modelos financieros. Además, si usa un lenguaje de programación, le permitirá conservar sus programas en ficheros del disco.

Asignación de nombres a los ficheros

Para almacenar un fichero en disco, hay que darle un nombre. Cada fichero se referencia con un único nombre llamado *especificador de fichero.* El especificador de fichero está dividido en dos partes: el *nombre del fichero* y la *extensión.* Cada fichero tiene que tener un nombre, pero la extensión es opcional. El nombre del fichero y la extensión se usan para describir el contenido del fichero. La combinación del nombre del fichero y la extensión tiene que ser única en el disco; es decir, un disco no puede tener dos ficheros con especificadores de ficheros idénticos. Cuando se está nombrando un fichero, la regla que hay que recordar es que el nombre del fichero puede tener ocho caracteres y la extensión tres caracteres.

El nombre del fichero y la extensión van siempre separados por un punto (.). Por ejemplo, PHONES.DAT es un especificador de fichero: PHONES es el nombre del fichero y DAT es la extensión. Otros ejemplos de especificadores de ficheros son ACCTS.BAS, DENTAL12.COM, y TIMEPRNT (está permitido tener un nombre de fichero sin extensión, como en el último ejemplo). Cuando le pida al MS-DOS que lea o escriba un fichero, tiene que darle el nombre del fichero y la extensión, si la tiene.

El término «nombre de fichero» se usa a menudo en los manuales de las computadoras, en lugar de «especificador de fichero». Si lee «nombre de fichero», recuerde que casi siempre se refiere a todo el especificador del fichero, incluyendo la extensión. La extensión se llama frecuentemente *tipo de fichero.*

Hay bastante libertad a la hora de dar nombre a los ficheros, pero recuerde las siguientes normas cuando elija los nombres y extensiones de los ficheros:

- No todos los caracteres del teclado están permitidos en los nombres de ficheros. Muchos de ellos se usan en otras partes del MS-DOS y se podrían confundir si apareciesen en un nombre de fichero. Los caracteres que puede y no puede usar al nombrar sus ficheros se muestran en la figura 2-3.
- Elija un nombre de fichero relacionado con el contenido del fichero. Por ejemplo, una carta dirigida a una persona llamada Holly puede llamarse HOLLYLET.TXT, o su base de datos de impuestos (en inglés, taxes) para 1984 puede llamarse TAX-84.DAT. Los nombres de ficheros apropiados hacen más fácil el conocer el contenido inmediatamente, sin tener que ver el contenido del fichero en la pantalla.
- El MS-DOS no permite nombres de ficheros iguales a los nombres de dispositivos que utiliza. (Los nombres de dispositivos son nombres para dispositivos hardware específicos, como la impresora). Los nombres que

PERMITIDOS

Todas las letras (sólo mayúsculas):

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Los números:

0 1 2 3 4 5 6 7 8 9

Muchos de los signos de puntuación, específicamente:

! @ # $ % ^ & ( ) { } - _ ´ ` ~

NO PERMITIDOS

No se pueden usar los siguientes caracteres:

, . / \ ¦ ? * ″ : ; [ ] + =

Figura 2-3. Caracteres permitidos y no permitidos en los nombres de ficheros.

no se pueden usar son AUX, COM1, COM2, CON, LST, LST1, LST2, LST3, PRN y NUL. Cuando sepa más de los dispositivos verá por qué el MS-DOS se confundiría si usted usara estos nombres como nombres de ficheros.

- Incluya una extensión en los especificadores de ficheros, aunque no sea obligatoria. Un nombre más detallado da más información cuando posteriormente se ve la lista de ficheros. La extensión también añade información al nombre del fichero que puede ser importante después. Por ejemplo, cuando usted escribe una carta a Edsel Corporation, en lugar de llamar al fichero EDSEL, puede llamarle EDSEL.LET o EDSEL.TXT. Si, posteriormente, usted crea otro fichero con cálculos de Edsel, puede llamarle EDSEL.DAT. La extensión le permite fácilmente diferenciarlo del otro fichero EDSEL.

- Nombre los ficheros relacionados con el mismo tema con nombres similares. Esto hace mucho más fácil buscar en el disco un fichero o un grupo de ficheros. Así, puede nombras las memorias dirigidas a Ms. Wong como WONG1.MEM, WONG2.MEM, y así sucesivamente.

- Use las extensiones estándar cuando sea posible. La tabla 2-2 lista algunas de las extensiones convencionales utilizadas en la industria de las microcomputadoras. Naturalmente, puede ignorar estas convenciones y nombrar sus ficheros como más le guste. Sin embargo, algunos programas asumirán que sus ficheros tienen ciertas extensiones y no serán capaces de encontrarlos si no las tienen.

- Evite el uso de algunas extensiones que tienen significados especiales para el MS-DOS (tales como COM, EXE y BAT). Si usa estas letras el MS-DOS puede asumir que estos ficheros son órdenes e intentar ejecutarlos. Esto podría causar resultados desastrosos. (A propósito, se puede decir qué ficheros son órdenes, programas o ficheros para tratamiento por lotes, mirando sus extensiones. Las órdenes tienen siempre las ex-

Tabla 2-2. Extensiones de ficheros más usuales

Extensión	Significado
ASM	Programa fuente en ensamblador
BAK	Copia de seguridad de otro fichero
BAS	Programa en BASIC
BAT	Fichero batch
BIN	Fichero binario usado por un programa
C	Programa fuente en lenguaje C
COB	Programa fuente en Cobol
COM	Programa
DAT	Fichero de datos
EXE	Programa
OBJ	Fichero objeto de un compilador
OVR	Fichero suplementario para un programa de aplicación
PAS	Programa fuente en Pascal
SYS	Programa especial del sistema
TEX	Fichero texto
TXT	Fichero texto

tensiones COM o EXE y los ficheros para tratamiento por lotes o ficheros «batch» tienen la extensión BAT).

- Aunque se pueden usar muchos signos de puntuación, la mayoría de la gente sólo usa unos cuantos. En general, utilizando el signo del dólar ($), el guión (-) y el subrayado (_) hay suficientes caracteres para definir un nombre. La figura 2-4 muestra algunos especificadores de ficheros válidos y algunos no válidos.

Válidos:	**Válidos, pero no recomendados:**
MODEMS.DOC	XX~^^XX
CHAP-3A.TXT	!!!!!!!!.@@@
PHONES.DAT	
10IDEAS	

No válidos:	**Causa:**
COMPUTERS.DOC	Nombre de fichero de más de 8 caracteres
CHAP-3A.TEXT	Extensión de más de 3 caracteres
PHONES.NAMES.DAT	Dos extensiones
10IDEAS?	Caracter no válido (?)

Figura 2-4. Ejemplos de especificadores de ficheros válidos y no válidos.

Utilización de caracteres comodín para grupos de ficheros

Se pueden utilizar algunas órdenes del MS-DOS para trabajar sobre un grupo de ficheros con nombres o extensiones similares. El agrupar ficheros cuando se utiliza una orden puede ahorrar gran cantidad de escritura en el teclado. El uso de un nombre común para un grupo de fichero es lo mismo que utilizar el apellido para designar a una familia. En lugar de decir «Quiero reunirme con Bob Patterson, con Jennifer Patterson, con Duane Patterson y con Randy Patterson», puede decir «Quiero reunirme con la familia Patterson». Similarmente, se pueden utilizar los nombres para encontrar a un grupo de personas: «Quiero encontrar a todas las personas que haya aquí, cuyo nombre sea Sally».

El empleo de un nombre para un grupo de ficheros puede ahorrar una gran cantidad de tiempo y de esfuerzo. Por ejemplo, usted puede querer transferir copias de 40 programas en BASIC desde A: hasta B:. Necesitaría dar 40 veces la orden COPY:

```
A>COPY PROG1.BAS B:

A>COPY PROG2.BAS B:
```

y así sucesivamente.

Para evitarle este enorme trabajo, el MS-DOS tiene dos caracteres especiales, que usted puede usar en el nombre del fichero, cuando quiere especificar más de un fichero: el signo de interrogación (?) y el asterisco (*). Estos caracteres se usan en especificadores de ficheros cuando se quiere que el MS-DOS actúe sobre un grupo de ficheros en lugar de sobre un fichero individual. En vez de emparejar los nombres de ficheros letra por letra, estos caracteres hacen que el MS-DOS busque los ficheros que tengan cualquier carácter en la parte del nombre del fichero que tiene el ? o el *. De la misma manera que en el poker los comodines pueden usarse para representar a cualquier carta, estos dos caracteres se pueden usar para representar cualquier carácter del teclado. Estos caracteres se llaman, apropiadamente, *caracteres comodín.*

El ? se utiliza para emparejar con cualquier carácter en la posición en que se encuentra, dentro del nombre del fichero.

Por ejemplo, si quiere copiar todos los ficheros cuyos nombres tienen cinco caracteres, los primeros cuatro son PROG y que tienen como extensión BAS, daría la orden

```
A>COPY PROG?.BAS B:
```

La orden copiaría PROG*1*.BAS, PROG*2*.BAS, PROG*A*.BAS, PROG*B*.BAS, etc., de la unidad A: a la unidad B:. No copiaría ni PROG10.

BAS ni PROGRAM.BAS, puesto que sus nombres tienen más letras que PROG?.BAS y el signo de interrogación sólo empareja con un carácter. Otro ejemplo es

```
A>COPY TAX?83.DAT B:
```

Esta orden copiaría TAXA83.DAT, TAX/83.DAT, TAX-83.DAT, etc., pero no copiaría TAXAB83.DAT ni TAXA.DAT.

Un segundo carácter comodín, el carácter *, emparejará con cualquier número de caracteres en la posición en la que aparece en el nombre del fichero. Esto significa que el número de letras que se emparejan no tiene importancia. También significa que se pueden emparejar ficheros con letras distintas en la posición que se ha especificado con el carácter *.

Por ejemplo, para copiar cualquier fichero cuyo nombre empieza con las letras PROG y cuya extensión es BAS, dé la orden

```
A>COPY PROG*.BAS B:
```

Esto empareja con PROG1.BAS, PROG1A.BAS, PROG9999.BAS y PROG2B2B.BAS.

Como podemos ver comparando este ejemplo con el anterior, usando el asterisco (a veces llamado estrella) se emparejan, al menos, tantos ficheros como usando el signo de interrogación.

Se puede usar más de un carácter comodín en una especificación de fichero. Por ejemplo, B???E.* empareja con los ficheros llamados BLARE.COM, B000E.123, B-D-E.TXT y BRAKE.

El uso de un carácter comodín hace mucho más fácil de resolver el problema primitivo (cómo copiar todos los programas BASIC de A: a B:). Para copiar cualquier fichero que tenga la extensión BAS, sin tener en cuenta el número de letras del nombre, usted debe reemplazar el nombre del fichero por un asterisco después de la orden COPY:

```
A>COPY *.BAS B:
```

Si, en vez de esto, quiere copiar todos los ficheros a B:, sin tener en cuenta el nombre ni la extensión, debe dar la orden

```
A>COPY *.* B:
```

Este es el mismo procedimiento que se utilizó previamente en este capítulo para hacer una copia del disquete del sistema.

Uso de nombres de dispositivos en las órdenes

Como hemos visto, la orden COPY se puede usar para copiar ficheros de un disco a otro. También se puede usar la orden COPY para mover información entre dispositivos. El MS-DOS proporciona un medio extremadamente fácil para comunicar con dispositivos tales como la impresora, las puertas para comunicaciones, el teclado y la pantalla.

Es altamente improbable que necesite utilizar nombres de dispositivos con la orden COPY, ya que la mayor parte de los programas que usan dispositivos tienen órdenes internas que comunican automáticamente con éstos. En caso de que necesite saber los nombres, éstos están catalogados y definidos en la tabla 2-3. Todos los nombres de dispositivos tienen tres caracteres (a veces seguidos por un número) y dos puntos (:) a continuación. Algunos de los dispositivos son sinónimos entre sí, como AUX: y COM1:, o LPT1: y PRN:.

Tabla 2-3. Nombres de dispositivos usados por el MS-DOS

Nombre y Sinónimo	Dispositivo
CON:	Consola. En realidad, es la combinación del teclado y la pantalla. La entrada se toma del teclado y la salida se visualiza en la pantalla.
AUX: y COM1:	La primera puerta de comunicaciones (puerta serie). También se conoce como Puerta adaptadora de comunicaciones asíncronas o Puerta RS-232. No todas las computadoras tienen esta puerta. Las puertas de comunicaciones segunda y tercera (si la computadora las tiene) se llaman COM2: y COM3:.
LPT1: y PRN:	La puerta de impresora en paralelo. A veces se conoce como Puerta Centronics o Puerta paralelo. No todas las computadoras tienen esta puerta. Puesto que muchas impresoras usan una puerta serie, puede que usted tenga que redirigir la entrada-salida hacia esta puerta con la orden MODE u otra orden similar proporcionada por el fabricante. Las puertas paralelo segunda y tercera (si la computadora las tiene) se llaman LPT2: y LPT3:.
NUL:	Dispositivo inexistente que se usa exclusivamente para probar programas de aplicación.

En lugar de usar el nombre de un fichero en la orden COPY, se usa el nombre del dispositivo. De la misma forma que un fichero, cualquier dispositivo se puede usar como entrada (lectura) o como salida (escritura). En realidad, puesto que la comunicación con estos dispositivos es similar a la copia de datos entre ficheros, el estudio de los nombres de dispositivos debe estar a continuación del estudio de los nombres de ficheros. La única orden con la que se pueden usar nombres de dispositivos es la orden COPY.

Por ejemplo, puede que usted quiera enviar un fichero a una computadora que está conectada a su puerta de comunicaciones, a través de un cable. Para hacerlo debe dar la orden

```
A>COPY MYFILE.TXT COM1:
```

Esto hace que el fichero MYFILE.TXT se copie del diquete de la unidad A: a la puerta de comunicaciones.

Otro ejemplo sería enviar el fichero a una impresora conectada a la puerta paralelo, para lo cual se daría la orden.

```
A>COPY MYFILE.TXT PRN:
```

UTILIZACION DE FICHEROS BATCH PARA COMBINAR ORDENES DEL MS-DOS

Ahora que usted ha estudiado un cierto número de órdenes del MS-DOS y se ha familiarizado con el método general para usar las órdenes, ya está preparado para aprender a ejecutar un grupo de ellas, una tras otra. Si tiene que ejecutar varias órdenes, puede ahorrar gran cantidad de tiempo, combinándolas en un fichero texto denominado *fichero batch* o *fichero para tratamiento por lotes.* Cuando se ejecuta un fichero batch se ejecutan las órdenes que contiene, de la misma forma que si se estuvieran tecleando en ese momento.

Escribir un fichero batch es igual que escribir un programa muy simple que dé al MS-DOS una lista de órdenes para ejecutar. Sin embargo, esto es mucho más sencillo que programar. Cuando usted está preparado para ejecutar las órdenes, le da al MS-DOS el nombre del fichero batch, en lugar del nombre de cada una de las órdenes.

En un fichero de este tipo, se puede utilizar cualquier orden o programa de los que se ejecutan normalmente a partir de la petición de orden A>. Combinando las órdenes de esta manera se ahorra bastante tiempo de mecanografía.

Por ejemplo, suponga que utiliza un programa de contabilidad llamado ACCT que crea un fichero llamado SALES.DAT. Usted ejecuta el programa

todos los días, y cada vez que ejecuta ACCT también utiliza la orden COPY para copiar el fichero SALES.DAT al fichero SALESNEW.DAT. Además, el programa ACCT requiere que se le dé, en la línea de la orden, el nombre de otros dos ficheros con los que trabaja, MAINACCT.DAT y NEWREPT.TXT. Sin usar un fichero batch se darían las siguientes órdenes

```
A>ACCT SALES.DAT MAINACCT.DAT NEWREPT.TXT

A>COPY SALES.DAT SALESNEW.DAT

A>
```

Puesto que siempre se escriben los tres nombres de ficheros y la segunda orden, se puede reducir bastante la escritura utilizando un fichero batch. Para crear el fichero se usa un editor de texto. Suponga que al fichero se le da el nombre DAILY.BAT:

DAILY.BAT

```
ACCT SALES.DAT MAINACCT.DAT NEWREPT.TXT
COPY SALES.DAT SALESNEW.DAT
```

Cada vez que vea el dibujo de una especie de carpetilla como ésta, significa que el contenido es un fichero MS-DOS. Ahora, cuando quiera ejecutar el fichero, usted dará una única orden, DAILY, al MS-DOS (no se necesita escribir la extensión BAT). Cuando el MS-DOS encuentre el fichero llamado DAILY.BAT, comenzará a ejecutar las órdenes que contiene:

```
A>DAILY

A>ACCT SALES.DAT MAINACCT.DAT NEWREPT.TXT

A>COPY SALES.DAT SALESNEW.DAT

A>
```

Observe que el MS-DOS primero encuentra el fichero (DAILY.BAT) y a continuación ejecuta cada línea contenida en él.

Cuando el MS-DOS ejecuta cada línea, la escribe en la pantalla.

Reglas para nombrar ficheros batch

Hay unas cuantas reglas que se deben seguir cuando se nombran ficheros batch:

- El nombre de un fichero batch tiene que incluir la extensión BAT para que el MS-DOS lo reconozca como fichero batch.
- No le dé un nombre que coincida con una orden a un fichero batch. Si lo hace, el MS-DOS ejecutará la orden en vez del fichero batch. Cuando se da una orden al MS-DOS, éste comprueba primero si es una orden interna; si no lo es, busca el nombre con una extensión COM, después con una extensión EXE y a continuación con una extensión BAT.
- Puesto que los ficheros batch son igual que los otros ficheros, se les aplican las mismas reglas en cuanto a los caracteres permitidos y no permitidos.

En un fichero batch se puede poner cualquier orden del MS-DOS y el fichero puede tener cualquier longitud. Cada orden tiene que estar en una línea separada, puesto que el MS-DOS lee las órdenes del fichero batch igual que como se le habrían introducido en respuesta a la petición de orden del MS-DOS.

El MS-DOS también proporciona órdenes especiales para usarlas sólo en los ficheros batch a fin de hacer aún más fácil el tratamiento por lotes. Estas órdenes especiales permiten cambiar discos entre medias de las órdenes, comprobar si un fichero particular existe, y realizar otras tareas útiles. Estas órdenes para ficheros batch están descritas en «Ordenes de ficheros batch» en el capítulo 3.

Utilización de argumentos dentro de los ficheros batch

Los ejemplos previos muestran cómo usar los ficheros batch cuando siempre se ejecuta el mismo conjunto de órdenes con los mismos argumentos. Pero ¿qué se puede hacer si se quiere que una orden del fichero batch tenga diferentes argumentos cada vez que se ejecuta dicho fichero? El MS-DOS permite utilizar la *sustitución de argumentos* para cambiar los argumentos cuando se ejecuta el fichero batch. Estos nuevos argumentos se usan entonces en las órdenes contenidas en el fichero batch. Incidentalmente, el MS-DOS no se preocupa del tipo de argumento que se le da; puede ser un nombre de fichero, el nombre de una unidad o cualquier otra cosa.

Para sustituir un argumento en la línea de orden de un fichero batch, se usa un signo de porcentaje y el número del argumento dentro del fichero batch. (Por ejemplo, el primer argumento se llamaría %1). El *signo de porcentaje* le dice al MS-DOS que reemplace el %1 en el fichero batch por el primer

argumento de la línea de orden, que reemplace el %2 en el fichero batch por el segundo argumento de la línea de orden, y así sucesivamente. Lo siguiente es un ejemplo de dónde puede ser útil la sustitución de argumentos.

Suponga que necesita copiar un grupo de ficheros de informes desde el disco A: bien al disco B: o al disco C:. Puede hacer dos tipos diferentes de ficheros batch, uno que copia los ficheros a B: y otro que los copia a C:. Sin embargo, para ahorrar aún más tiempo, puede escribir un fichero batch que utiliza sustitución de argumentos. El fichero batch RPTCOPY.BAT podría ser

RPTCOPY.BAT

```
COPY *.RPT %1
```

Observe que en este fichero batch el segundo argumento después de la orden COPY es %1. Cada vez que use este fichero batch, tiene que dar como argumento el nombre del disco sobre el cual quiere copiar los ficheros. El MS-DOS sustituirá entonces en la orden COPY, contenida en el fichero batch, el nombre del dispositivo que usted le indique.

Para ejecutar el fichero batch RPTCOPY con C: como argumento, escriba

```
A>RPTCOPY C:

A>COPY *.RPT C:
        5 File(s) copied

A>
```

Cuando el MS-DOS ve el % 1 en el fichero RPTCOPT, lo reemplaza por C: puesto que es el primer argumento en la línea de orden. Puede usar el mismo fichero batch para copiar en el disco B:

```
A>RPTCOPY B:

A>COPY *.RPT B:
        5 File(s) copied

A>
```

Si tiene varias órdenes que utilizan el mismo argumento, cada vez que ejecuta el fichero batch, puede usar un argumento con signo de porcentaje, más de una vez.

Por ejemplo, suponga que usted ejecuta siempre la orden CALCBEG junto con la orden CALC-END. Cada una de estas órdenes lleva el mismo argumento, un nombre de fichero de datos. Uste puede escribir un fichero CALCBOTH.BAT para ejecutar los programas CALC-BEG y CALC-END con el mismo argumento.

CALCBOTH.BAT

```
CALC-BEG %1
CALC-END %1
```

Cuando ejecute este fichero con FY1982.DAT como argumento, verá lo siguiente:

```
A>CALCBOTH FY1982.DAT

A>CALC-BEG FY1982.DAT

A>CALC-END FY1982.DAT

A>
```

Si se quiere substituir varios argumentos en una orden, se pueden usar más signos de porcentaje: %2, %3, %4, etc. En realidad, el MS-DOS permite dar hasta nueve argumentos a un fichero batch. Igual que antes, si usted usa un segundo argumento, el MS-DOS reemplazará cada %2, presente en el fichero batch, por el segundo argumento, reemplazará cada %3 por el tercer argumento, y así sucesivamente.

La capacidad del MS-DOS para incluir muchos argumentos en un fichero batch es una herramienta valiosa. Con ella, se pueden crear ficheros batch que realicen tareas complejas.

Suponga que necesita dar un nombre único al segundo y tercer fichero en el programa de contabilidad del fichero ejemplo DAILY.BAT, utilizado previamente. Debería comenzar reescribiendo ese fichero de forma que incluya dos argumentos a sustituir:

DAILY.BAT

```
ACCT SALES.DAT %1 %2
COPY SALES.DAT SALESNEW.DAT
```

A continuación, dé la orden DAILY y los dos nombres de ficheros:

```
A>DAILY RECEIVE.DAT RCVREPT.TXT

A>ACCT SALES.DAT RECEIVE.DAT RCVREPT.TXT

A>COPY SALES.DAT SALESNEW.DAT

A>
```

Ahora que se ha familiarizado con la sustitución de argumentos en los ficheros batch, entenderá por qué se debería evitar el uso de signos de porcentaje en los nombres de ficheros (como se observa en la figura 2-3). Si tiene un especificiador de fichero en su programa batch, que contiene un signo de porcentaje (tal como N%X.DAT), tiene que recordar que debe ponerle otro signo de porcentaje (N%%X.DAT). Este segundo signo de porcentaje le dice al MS-DOS que el primer signo forma parte del nombre del fichero.

El fichero AUTOEXEC.BAT

Hay un tipo de fichero batch que no se ha tratado todavía. Este fichero batch se ejecuta automáticamente (o se «autoejecuta») cada vez que se pone en marcha la computadora, y se llama fichero AUTOEXEC.BAT. Este fichero se puede utilizar para ejecutar ciertos programas automáticamente o para imprimir los resultados de los programas ejecutados el día anterior, tan pronto como se realiza la carga inicial de la computadora.

Para escribir un fichero AUTOEXEC.BAT, simplemente se crea un fichero batch (igual que los que ya hemos visto) y se le da el nombre de AUTOEXEC.BAT. Este fichero puede contener todas las órdenes que usted quiera. A continuación, cada vez que se haga la carga inicial del MS-DOS, éste buscará el fichero AUTOEXEC.BAT y, si lo encuentra, automáticamente ejecutará las órdenes. Incidentalmente, éste es el fichero especial, mencionado al comienzo del capítulo, que evita que el MS-DOS pregunte la fecha y la hora.

Las órdenes que actualizan los indicadores del sistema (tales como la fecha y la hora) se colocan generalmente en el fichero AUTOEXEC.BAT. También se pueden incluir algunos paquetes de software, cuando se pretende que se carguen en memoria inmediatamente después de encender el sistema. Un ejemplo de un fichero AUTOEXEC.BAT típico se da al final de la sección «Ordenes de ficheros batch» en el capítulo 3.

Enlace de ficheros batch

Se puede incluir el nombre de un fichero batch dentro de otro fichero batch, con tal de que se coloque al final del fichero. Esto quiere decir que los ficheros batch se pueden enlazar. Por ejemplo, se puede escribir un fichero batch que primero ejecuta los programas de inventario y a continuación, enlaza con otro fichero batch que imprime la salida. (La ventaja de separar las órdenes de impresión en un segundo fichero batch es que así se puede usar este segundo fichero él solo).

Recuerde, sin embargo, que sólo se pueden enlazar programas por la última línea del fichero batch. El MS-DOS es incapaz de volver a un fichero batch, una vez que ha enlazado con otro, aunque existe una excepción. Este problema y su solución están completamente explicados en «La orden COMMAND» en el capítulo 3.

UNA SESION DE EJEMPLO CON EL MS-DOS

Ahora usted sabe casi todo lo que necesita para comenzar a usar el MS-DOS igual que un profesional. En esta sección se presenta un ejemplo de una sesión típica que podría usted tener con el MS-DOS.

Suponga que sus tareas para ese día son ejecutar un paquete de generación de informes, que forma parte del sistema de control de inventario, y escribir una memoria con los resultados del informe. También debe imprimir la memoria y hacer una copia en un disquete tanto de la memoria como del informe. Estas tareas se presentarán paso a paso.

Paso 1: Uso de un fichero batch para ejecutar el programa

Lo primero, usted quiere ejecutar el programa que hace el informe del inventario. Ejecutar este programa es fácil porque usted ya ha escrito un fichero batch (llamado INVRPT) para manejar las tres órdenes diferentes que da usualmente. El fichero INVRPT contiene

INVRPT.BAT

```
INVMAKE %1 %2 INVKEY.TXT
INVFILE INVKEY.TXT
INVOUT %1 %2 INVKEY.TXT
```

Todo lo que necesita decirle al fichero lote son las fechas inicial y final para el informe.

```
A>INVRPT 8/1/84 8/15/84

A>INVMAKE 8/1/84 8/15/84 INVKEY.TXT

A>INVFILE INVKEY.TXT

A>INVOUT 8/1/84 8/15/84 INVKEY.TXT
The report is saved in file INVRPT93.TXT

A>
```

El último mensaje («El informe se ha guardado en el fichero...») lo ha escrito el programa INVOUT. Este programa hipotético le dice el nombre del fichero que ha producido y guardado en disco (en este caso, INVRPT93.TXT). Cuando acaba el programa vuelve la petición de orden A > .

Paso 2: Análisis del informe

El siguiente paso es analizar el informe que se acaba de producir. Usted puede o bien mirarlo en la pantalla o escribirlo por la impresora. Para visualizar el informe en la pantalla utilice la orden TYPE.

```
A>TYPE INVRPT93.TXT

          Inventory Report for 8/1/84 to 8/15/84

Parts that need to be reordered by 8/20/84:

Item            On hand         Reorder level       Price
----------------------------------------------------------
CF3499            8                   12            14.50
CF5000           52                   60             1.19
  .               .                    .               .
  .               .                    .               .
  .               .                    .               .
```

Si el informe es más largo de lo que esperaba, puede imprimirlo con la orden PRINT. La orden PRINT es igual que la orden TYPE, salvo que saca el fichero por la impresora en vez de por la pantalla.

```
A>PRINT INVRPT93.TXT

          A:INVRPT93.TXT is currently being printed

A>
```

Paso 3: Escritura de la memoria

Después de evaluar el informe, usted usa su programa procesador de texto para escribir una memoria detallando sus recomendaciones. Puesto que cada procesador de texto es diferente, este libro no intenta explicar cómo escribir la memoria.

Suponga que está ejecutando un procesador de texto llamado TEXTP y que quiere editar el fichero INVRPT93.TXT. Usted daría la orden

```
A>TEXTP INVRPT93.TXT
```

Usted añade texto al informe que se había producido y posiblemente reordena los resultados. Cuando acaba la memoria, quiere escribirla por la impresora. Muchos procesadores de texto imprimen la memoria, si se les pide, pero otros no. En este caso, TEXTP no lo hace, así que usted tiene que volver al MS-DOS. Guarda la memoria con el nombre NEWINV.MEM, usa una orden del TEXTP para volver al MS-DOS y, a continuación, da la orden PRINT.

```
A>PRINT NEWINV.MEM

          A:NEWINV .MEM is currently being printed

A>
```

Paso 4: Copia de la memoria y del informe

Recuerde que en este trabajo, usted quería una copia en disco de la memoria y del informe. Para hacerlo, ponga un disco formateado en la segunda unidad (B:) y asegúrese de que está en blanco usando la orden DIR.

```
A>DIR B:

 Volume in Drive B has no label
 Directory of B:\

File not found

A>
```

El mensaje anterior al de «File not found» (fichero no encontrado) variará según la versión del MS-DOS que se esté ejecutando (esto aparece en la versión 2). Sin embargo, realmente no importa, porque se ha encontrado lo que se quería saber: no hay nada en el disco.

Ahora copie los ficheros que contienen el informe y la memoria:

```
A>COPY INVRPT93.TXT B:
        1 File(s) copied

A>COPY NEWINV.MEM B:
        1 File(s) copied

A>
```

Recuerde que la orden COPY lleva dos argumentos: el origen de los datos y el destino de los datos. En estos casos, el origen es el fichero en A:, la unidad por defecto, y el destino es la unidad B:. Para estar seguro de que copió todo correctamente, compruebe el contenido de B: con la orden DIR:

```
A>DIR B:

 Volume in Drive B has no label
 Directory of B:\

INVRPT93 TXT    8110     8-20-84  10:42p
NEWINV   MEM     631     8-20-84  11:20p
       2 File(s)   151515 bytes free

A>
```

La orden DIR le da el nombre del fichero, extensión del fichero, número de bytes en el fichero y la fecha y hora en que fué creado. También le dice cuántos ficheros hay en el disco y la cantidad de espacio que queda hasta llenar el disco («bytes libres»).

Ahora, usted ha terminado sus tareas. Debería encontrarse a gusto dando órdenes al MS-DOS y contestando las preguntas que le hace el programa. Usted está preparado para aprender las otras muchas órdenes del MS-DOS que están descritas en el próximo capítulo.

3

ORDENES DEL MS-DOS

Organización del capítulo
Cómo leer la sintaxis de una orden
Parada de una orden
Ordenes de mantenimiento de ficheros
Ordenes de salida de ficheros
Ordenes de mantenimiento de discos
Indicadores del sistema
Ordenes de mantenimiento de caminos
Ordenes de ficheros batch
Otras órdenes
Herramientas de programación

Este capítulo describe las órdenes del MS-DOS incluidas en su sistema y le dice cómo usarlas. La mayoría de las descripciones de órdenes contienen ejemplos que muestran cómo introducir las órdenes. Estos ejemplos son hipotéticos; por ejemplo, la mayor parte contienen nombres de ficheros que no están presentes en sus discos. Debería usarlos como prototipos generales de cómo usar una cierta orden para un cierto propósito.

Además de las órdenes estándar del MS-DOS, este capítulo también cubre muchas órdenes que IBM ha añadido al PC-DOS. Otros fabricantes han copiado muchas de estas órdenes específicas de IBM, así que puede encontrarlas en computadoras que no son IBM.

Aunque algunas versiones del MS-DOS tienen órdenes ligeramente diferentes sólo varían dentro de los límites de las principales versiones. Por esta razón, las órdenes en la versión 1.0 realizan la misma función que en la siguiente versión principal, la versión 1.1, pero pueden ser diferentes en la versión 2. Este capítulo trata las órdenes utilizadas en ambas versiones; en las órdenes nuevas que están incluidas sólo en la versión 2, se advierte este hecho al principio de la orden. El apéndice B proporciona más información sobre las diferencias entre las órdenes de las versiones 1 y 2.

Su computadora puede que tenga MS-DOS y además programas de aplicación tales como procesador de texto o programa de hoja de cálculo. Puesto que el MS-DOS no contiene programas de aplicación, éstos no se explican aquí. Los manuales de estos programas deben estar incluidos con ellos. Si encuentra discos de programas sin ninguna documentación (es decir, un manual que explique cómo usar el programa), pídala al distribuidor de su computadora. Los programas de aplicación que se ejecutan con el MS-DOS se tratan en el capítulo 6.

ORGANIZACION DEL CAPITULO

Todas las órdenes del MS-DOS, que aparecen en este capítulo, se presentan en grupos funcionales generales. Por ejemplo, todas las órdenes utilizadas para mantener los ficheros (como copiar y borrar) se agrupan bajo el encabezamiento «Ordenes de mantenimiento de ficheros». La figura 3-1 muestra todos estos grupos funcionales, las órdenes de cada grupo y la página en la que se describe cada orden.

Si está interesado en utilizar este capítulo como referencia, el apéndice A le será de gran utilidad. El apéndice A proporciona una lista alfabética de todas las órdenes, su sintaxis y el grupo funcional dentro del cual están descritas en este capítulo.

La descripción de cada orden incluye la siguiente información:

- *Recuadro del perfil de la orden.* Incluye todos los nombres abreviados de la misma orden. También indica si la orden se introdujo en la versión 2 del MS-DOS, si es interna o externa y si es específica de IBM.
- *Usos más comunes.* Esta sección muestra cómo usar la orden y todos sus argumentos de distintas maneras. Contiene la mayoría de los ejemplos dados para cada orden. Una cuantas órdenes también contienen una sección llamada «Otros usos» que describe los usos menos prácticos de la orden.
- *Reglas.* Algunas órdenes tienen reglas definidas relativas a cuándo pueden usarse y cuándo no. Encontrará esas reglas en esta sección.

Ordenes del MS-DOS		Página donde encontrar la orden
Ordenes de mantenimiento de ficheros		
ERASE	Suprime ficheros de un disco	52
RENAME	Cambia de nombre a los ficheros	54
COPY	Hace copias de ficheros	57
COMP	Compara ficheros	63
EDLIN	Edita ficheros texto	65
Ordenes de salida de ficheros		
TYPE	Visualiza un fichero en la pantalla	67
PRINT	Imprime un fichero en la impresora	68
MORE	Visualiza un fichero en la pantalla con pausas	72
Ordenes de mantenimiento de discos		
DIR	Visualiza la lista de ficheros de un disco	74
FORMAT	Prepara un disco para el MS-DOS	78
SYS	Graba los ficheros del sistema y las pistas de carga inicial en un disquete	82
CHKDSK	Examina y repara discos	84
RECOVER	Repara ficheros con sectores en mal estado	89
DISKCOPY	Copia un disquete entero	91
DISKCOMP	Compara dos disquetes	94
FDISK	Realiza el mantenimiento en el IBM PC/XT	96
BACKUP	Hace copias de seguridad del disco duro	97
RESTORE	Hace copias de los ficheros de seguridad	97
Indicadores del sistema		
DATE	Pone la fecha	99
TIME	Pone la hora	100
VERIFY	Dice al MS-DOS que haga una doble comprobación cuando escriba en los discos	101
MODE	Cambia los parámetros de comunicación	103
VER	Visualiza el número de la versión del MS-DOS	108
VOL	Visualiza el rótulo de un disco	109
PROMPT	Cambia la petición de orden del MS-DOS	110
BREAK	Hace que el MS-DOS compruebe más a menudo si hay interrupciones de usuario	112
ASSIGN	Cambia las letras asignadas a los discos	114
CTTY	Cambia la consola a una puerta serie	116
SET	Cambia las cadenas de entorno	117

Figura 3-1. Grupos de órdenes en este capítulo

	Ordenes del MS-DOS	Página donde encontrar la orden
Ordenes de mantenimiento de caminos		
MKDIR	Añade subdirectorios a un disco	120
CHDIR	Cambia el subdirectorio por defecto	122
RMDIR	Suprime subdirectorios de un disco	124
TREE	Saca una lista de subdirectorios	126
PATH	Cambia la lista de búsqueda de órdenes	128
Ordenes de ficheros batch		
ECHO	Escribe un mensaje en la pantalla	132
PAUSE	Espera a que se pulse una tecla	133
IF	Ejecuta órdenes según una decisión	135
FOR	Repite una orden para muchos argumentos	139
GOTO	Salta a una parte diferente dentro del fichero batch	142
REM	Pone un comentario en un fichero batch	145
SHIFT	Mueve los argumentos de la línea de la orden	146
COMMAND	Ejecuta otro programa	147
Otras órdenes		
FIND	Busca texto en un fichero	150
SORT	Ordena un fichero texto	153
CLS	Borra la pantalla	155
GRAPHICS	Permite imprimir pantallas gráficas	155
Herramientas de programación		
LINK	Combina ficheros objeto	156
DEBUG	Examina y modifica ficheros binarios	158
EXE2BIN	Convierte ficheros EXE en ficheros COM	158

Figura 3-1. Grupos de órdenes en este capítulo *(continuación)*

- *Advertencias y errores frecuentes.* Esta sección lista los problemas que son frecuentes en cada orden específica; también proporciona advertencias que pueden ayudarle a evitar grandes problemas. (El capítulo 5, «Tratamiento de incidentes», explica cómo volver a la normalidad después de los negativos resultados de algunos errores graves).
- *Sintaxis.* La *sintaxis* es una disposición ordenada de los argumentos en la línea de la orden. Esta sección da la sintaxis completa para utilizar cada orden. La sección de sintaxis usualmente contiene mucha más información de la que usted necesita inmediatamente para usar la orden, pero esta información es útil para entender exactamente cómo dar la orden. Las sintaxis corrientes se dan en la sección «Usos más comunes».

COMO LEER LA SINTAXIS DE UNA ORDEN

Para dar una orden, usted tiene que aprender a escribir un poco del lenguaje del MS-DOS. Para que el MS-DOS sea capaz de leer la orden, usted necesita colocar el nombre de la orden y sus argumentos siguiendo la sintaxis de la orden. La sintaxis puede ser corta y simple o larga y compleja. Especifica todo lo que la orden necesita para realizar su trabajo. Si lee cuidadosamente la sintaxis y proporciona toda la información en los argumentos, usted puede estar seguro de que la orden se ejecutará sin problemas.

Los siguientes tres conceptos son importantes para saber cómo interpretar las notaciones sintácticas en este libro.

- *Símbolos especiales.* Este libro usa símbolos especiales para indicar los diferentes tipos de argumentos. Estos símbolos pueden verse en la tabla 3-1. Un argumento que no tenga símbolos especiales a su alrededor es obligatorio para esa orden. Como veremos, la mayoría de los argumentos son opcionales. Sería bueno que usted se familiarizase con estos símbolos especiales, ya que se usan tanto en este libro como en otros muchos libros de computadoras.
- *Argumentos en letra cursiva.* Estos argumentos, también llamados *argumentos genéricos,* se utilizan para indicar que el argumento puede tomar muchos valores. Por ejemplo, en la sintaxis «TYPE *especifichero*», *especifichero* indica que se puede usar cualquier nombre de fichero o especificador de fichero. El argumento *especifichero* es una palabra inventada que se usa para señalar dónde debe estar un nombre de fichero. Algunos de los argumentos genéricos utilizados a lo largo de este capítulo se muestran en la tabla 3-2; otros argumentos específicos de algunas órdenes, se describen cuando aparecen, dentro de la sintaxis individual de las órdenes.
- *Argumentos de una letra.* Muchas órdenes tienen argumentos que consisten en una letra precedida por una barra (por ejemplo, /V). Es

Tabla 3-1. Símbolos utilizados en la sintaxis de las órdenes

Símbolo	Descripción
[ ]	Los corchetes indican que todo lo que hay dentro es opcional.
{ }	Las llaves indican que hay que elegir exactamente uno de los valores posibles, situados entre las llaves. Cada valor posible está separado del siguiente por una barra vertical (|).
...	Los puntos suspensivos indican que se puede repetir de nuevo el último conjunto de argumentos.

Tabla 3-2. Argumentos genéricos en la sintaxis de las órdenes

Argumento genérico	Descripción
d:	Descriptor de disco, como por ejemplo A: o B:.
camino	Nombre del camino asociado al fichero.
nomfichero	Nombre de un fichero (o grupo de ficheros), excluyendo la extensión. Se pueden usar nombres de ficheros con caracteres comodín a no ser que la descripción de la orden diga otra cosa.
ext	Extensión de un fichero o grupo de ficheros. Se pueden usar caracteres comodín, a no ser que se diga lo contrario en la descripción de la orden.
especifichero	Especifica el fichero que contiene, al menos el nombre del fichero. Así, la sintaxis para un *especifichero* es [*d:*] [*camino*]*nomfichero*[.*ext*]
n	Un número. Por ejemplo, COM*n*: puede significar COM1:, COM2:, etc.
cadena	Grupo de caracteres. Por ejemplo, «xxyyz» es una cadena de cinco caracteres; «Hi there» es una cadena de ocho caracteres.

importante darse cuenta de que estos argumentos tienen distintos significados en distintas órdenes. Este uso de la misma letra con significados diferentes puede ser algo confuso. Usted debería comprobar siempre que el argumento de una letra que está utilizando es correcto para esa orden. (Algunos manuales llaman a estos argumentos de una letra *opciones*.) Cuando introduzca estos argumentos, asegúrese de que utiliza la barra hacia adelante (/) y no la barra hacia atrás (\).

En los siguientes ejemplos, se puede ver la importancia de comprender la sintaxis de una orden.

El primer ejemplo es

VERIFY [{OFF |ON}]

Los corchetes indican que la orden VERIFY puede o bien no tener argumentos o tener un único argumento; las llaves indican que el argumento puede ser OFF u ON. De este modo, la orden tiene que darse como VERIFY, VERIFY OFF o VERIFY ON.

El siguiente ejemplo es

DIR [*d*:]

Con esta sintaxis, la orden DIR puede darse o bien ella sola o con un descriptor de disco como B:.

El tercere ejemplo es

TYPE *especifichero*

El argumento *especifichero,* que habitualmente es el nombre de un fichero, es obligatorio en esta orden. Recuerde que *especifichero* puede también contener el nombre del disco o del subdirectorio (los subdirectorios se explican en el capítulo 4).

PARADA DE UNA ORDEN

Después de comenzar a ejecutar una orden, usted puede querer pararla. Esto puede suceder si usted descubre que la orden está escribiendo información que no necesita. Por lo general, puede parar la orden manteniendo pulsada la tecla CONTROL y pulsando simultáneamente la tecla C. Esto le indica al MS-DOS que usted quiere parar la orden. Después de pulsar CONTROL-C, la pantalla visualizaría la petición de orden A>.

Pulsar CONTROL-C funciona en la mayor parte de los casos, pero, desafortunadamente, no en todos los casos. Algunas computadoras usan la *tecla BREAK* en lugar de CONTROL-C. Otras computadoras, tales como el IBM PC requieren que se pulsen simultáneamente la tecla CONTROL y la tecla BREAK.

Si necesita parar una orden y el método que acabamos de describir no funciona, algunas computadoras tienen una *tecla de reset* que puede pulsar en último caso. Otras máquinas utilizan una serie de teclas. Por ejemplo, en el IBM PC para hacer el reset (reinicialización) hay que mantener pulsadas simultáneamente las teclas ALT, CONTROL y DELETE. Al pulsar la tecla de reset, le dice a la máquina que deje de hacer lo que esté haciendo, sin importar las consecuencias.

El uso del reset es una forma efectiva de parar una orden, pero *no* es recomendable. Usted puede dañar seriamente la información de los discos si la computadora estaba escribiendo información cuando pulsó la tecla de reset. Si necesita usar el reset, debería verificar después que todos los ficheros del disco están intactos. Puede hacerlo con la orden CHKDSK, descrita más adelante en este capítulo.

A veces se quiere que un programa se pare sólo temporalmente sin volver a la petición de orden A>. Si una orden está escribiendo una gran cantidad de información en la pantalla, por lo general usted puede parar el proceso pulsando una tecla (o combinación de teclas). Puesto que esta combinación

varía según la computadora que usted tenga, debería leer los manuales de su computadora para encontrar qué tecla o teclas paran un programa. (En el IBM PC, es la combinación de las teclas CONTROL y NUMLOCK). El pulsar la tecla RETURN se suele reanudar la ejecución de la orden.

ORDENES DE MANTENIMIENTO DE FICHEROS

Esta sección trata las siguientes órdenes:

ERASE (también DEL)	*Elimina ficheros de un disco*
RENAME (también REN)	*Da nuevos nombres a los ficheros*
COPY	*Hace copias de ficheros*
COMP	*Compara ficheros*
EDLIN	*Edita ficheros texto*

Estas órdenes del MS-DOS se usan para crear y manipular ficheros. Las tres primeras se usan muy a menudo; seguramente usará las dos últimas muy pocas veces, si es que las usa.

Orden ERASE
Orden DEL
Interna

La orden ERASE elimina un fichero o un grupo de ficheros del directorio de un disco. La orden DEL es idéntica a la orden ERASE.

Usos más comunes

Hay varias razones para usar la orden ERASE. Si uno de sus programas ha creado un fichero inútil, o si tiene un fichero que ya no necesita en un disco, puede eliminarlo con la orden ERASE. Puesto que la capacidad de almacenamiento es limitada, es importante borrar viejos ficheros y liberar espacio para nuevos ficheros. Sin embargo, recuerde que, una vez que se borra un fichero, es difícil o imposible recuperarlo. (Algunos programas de utilidades, que están a la venta, pueden recuperar en algunos casos ficheros borrados. Ver «Utilidades de conservación de discos» en el capítulo 5).

La sintaxis corriente de la orden ERASE es

ERASE *especifichero*

El proceso de borrado de un fichero es simple, necesitando solamente el nombre del fichero que se quiere eliminar.

Por ejemplo, para borrar un fichero llamado BOSTON88.TST en la unidad por defecto, escriba la orden

```
A>ERASE BOSTON88.TST
```

o:

```
A>DEL BOSTON88.TST
```

El fichero se eliminará del disco y del directorio.

Se pueden usar caracteres comodín con la orden ERASE. De esta forma, se pueden borrar varios ficheros a la vez.

Por ejemplo, para borrar todos los ficheros con la extensión BAK (que se usa para las copias de seguridad), dé la orden:

```
A>ERASE *.BAK
```

Si quiere borrar todos los ficheros de un disco, la orden ERASE le pedirá una confirmación antes de hacerlo.

```
A>ERASE *.*
Are you sure (Y/N)?Y
```

Cuando vea este mensaje, cuyo significado es, ¿está usted seguro? ¿Sí (Yes) o No?, piense cuidadosamente en los contenidos de los ficheros. Una vez que se borra un fichero, no hay ningún método fácil para recuperarlo; en realidad puede que no haya forma de recuperarlo.

Reglas

Usted tiene que decirle a la orden ERASE qué ficheros quiere borrar. Si se le olvida poner el nombre del fichero, el MS-DOS le dará el mensaje «Invalid number of parameters» (Número de parámetros no válido).

Si especifica el disco en el que está el fichero pero olvida el nombre del fichero, el MS-DOS escribirá «File not found» (Fichero no encontrado). Esto es porque, como precaución, la orden ERASE necesita que cuando se le da un especificador de disco, se le dé también un nombre de fichero.

Advertencias y errores frecuentes

Obviamente, tiene que tener cuidado de no borrar ficheros importantes. Si otras personas usan su computadora, sea prudente antes de borrar ficheros. Aunque usted pueda creer que un fichero no tiene importancia o que no sirve para nada, otra persona puede necesitar el fichero para una aplicación que usted no conoce.

Si usted está usando caminos, tenga especial cuidado con la orden ERASE. Si da el nombre de un subdirectorio como argumento de la orden ERASE, el MS-DOS borrará todos los ficheros de ese subdirectorio. Esto es equivalente a pasar al subdirectorio y dar la orden ERASE *.*. Los caminos y los subdirectorios se tratan en la sección de este capítulo «Ordenes de mantenimiento de caminos». También se describen en el capítulo 4.

Sintaxis

La sintaxis de la orden ERASE es

ERASE *especifichero*

Hay que dar a la orden ERASE un *especifichero* (especificador de fichero). El *especifichero* puede contener caracteres comodín.

Orden RENAME
Orden REN
Interna

La orden RENAME cambia el nombre de un fichero sin cambiar su contenido. La orden REN es idéntica a la orden RENAME.

Usos más comunes

Hay unas cuantas razones por las que usted puede querer cambiar el nombre de un fichero. El contenido de un fichero puede haber cambiado tanto que el nombre antiguo ya no es adecuado, o usted puede darse cuenta de que accidentalmente ha dado nombres similares a dos ficheros. Ahora puede cambiar el nombre a uno de los ficheros para evitar confusión. La sintaxis usual de la orden es

RENEME *especifichero1 especifichero2*

donde *especifichero1* es el nombre original del fichero y *especifichero2* es el nombre que se le quiere dar.

Por ejemplo, para cambiar el nombre de SALES.DAT a US-SALES.DAT, dé la orden

```
A>RENAME SALES.DAT US-SALES.DAT
```

o bien

```
A>REN SALES.DAT US-SALES.DAT
```

Puede usar la orden RENAME en conjunción con la orden ERASE para deshacerse de la versión actual de un fichero y reemplazarlo por una versión anterior. Puede que quiera hacer esto si revisa un fichero y decide que la versión original era más apropiada.

Por ejemplo, asuma que quiere cambiar el nombre de la copia de seguridad de un programa de facturación llamado BILLPROG.COB. Usted está usando un editor de texto que guarda una copia de seguridad con el mismo nombre que el fichero original, pero con la extensión BAK. Usted daría las siguientes órdenes

```
A>ERASE BILLPROG.COB
A>RENAME BILLPROG.BAK BILLPROG.COB
```

Si simplemente quiere intercambiar los nombres de dos ficheros, necesitará utilizar un nombre temporal, porque la orden RENAME no permite cambiar el nombre a los dos ficheros a la vez. Puede que usted quiera intercambiar dos nombres de ficheros si ha revisado un fichero y a continuación ha decidido usar la versión original, pero quiere guardar una copia de las revisiones para su uso posterior.

Por ejemplo, para intercambiar los nombres de los ficheros SALEPROG.-COB y SALEPROG.BAK, escriba las siguientes órdenes:

```
A>RENAME SALEPROG.COB TEMPFILE.COB
A>RENAME SALEPROG.BAK SALEPROG.COB
A>RENAME TEMPFILE.COB SALEPROG.BAK
```

En este ejemplo, TEMPFILE.COB es un nombre de fichero temporal.

Si tiene dos o más ficheros en un disco que comparten o bien el nombre o bien la extensión, puede usar caracteres comodín para cambiar el nombre de estos ficheros como grupo.

El primer ejemplo cambia los nombres de todos los ficheros con el nombre ACCOUNTS al nombre ACC.

```
A>RENAME ACCOUNTS.* ACC.*
```

El segundo ejemplo cambia los nombres de todos los ficheros con la extensión TXT a la extensión TEX.

```
A>RENAME *.TXT *.TEX
```

Reglas

No se puede usar RENAME para cambiar los nombres de un grupo de ficheros que están en discos diferentes; primero hay que usar la orden COPY para llevar los ficheros al mismo disco.

No se puede cambiar el nombre de un fichero a otro nombre que ya exista en el disco. Si lo intenta, el MS-DOS dará el mensaje «Duplicate file name or file not found» (Nombre de fichero duplicado o fichero no encontrado).

Advertencias y errores frecuentes

El error más frecuente consiste en olvidar el orden de los dos nombres de ficheros. RENAME requiere que el nombre antiguo esté el primero y el nuevo el segundo. (Los que conocen el sistema operativo CP/M se darán cuenta que esto es lo contrario del orden exigido en el CP/M).

Si varias personas usan su computadora, asegúrese de que cuando cambia el nombre a una orden o a un fichero, todos los que usan ese fichero conocen el nuevo nombre. Cambiar el nombre de un fichero sin previo aviso probablemente causará una confusión innecesaria.

No cambie los nombres de los ficheros especiales usados por programas de aplicación. El programa WordStar, por ejemplo, requiere que esté en el disco el fichero suplementario WSOVLY1.OVR. Si cambia el nombre de este fichero no hay forma de que el WordStar conozca el nuevo nombre.

Sintaxis

La sintaxis de la orden RENAME puede ser una de las dos siguientes:

RENAME *especifichero1 especifichero2*

o bien

RENAME *especifichero1 nomfichero2*[*.ext2*]

En la primera forma, *especifichero1* es el nombre del fichero que se quiere cambiar y *especifichero2* es el nuevo nombre. En la segunda forma *nomfichero2.ext2* es el nuevo nombre para el fichero (si no incluye la extensión, el MS-DOS usará la misma extensión que tenía el fichero original). Observe que el nombre antiguo se da el primero.

Orden COPY
Interna

La orden COPY permite copiar ficheros de un disco a otro. También permite hacer una copia de un fichero en el mismo disco (pero con un nombre diferente), copiar información en los dispositivos y de los dispositivos y combinar muchos ficheros en uno sólo.

Usos más comunes

La orden COPY es una de las órdenes usadas más frecuentemente en el MS-DOS. Se usa principalmente para hacer copias de ficheros de un disco a otro.

La primera sintaxis usual de la orden COPY es

COPY *origen destino*

Aquí, *origen* es el nombre del fichero que usted quiere copiar y *destino* es donde usted quiere copiarlo.

Por ejemplo, si quiere copiar el fichero OPENACCT.DAT de un disco en la unidad A: a un disco en la unidad B:, debe dar la orden

```
A>COPY OPENACCT.DAT B:
        1 File(s) copied
```

El fichero OPENACCT.DAT se ha copiado en la unidad B: con el mismo nombre.

La orden COPY siempre le dice cuántos ficheros ha copiado. Observe que no hace falta que dé el nombre del fichero en el disco de destino, sino sólo el nombre de la unidad. Esto se debe a que la orden COPY asume que usted quiere que el nuevo fichero tenga el mismo nombre.

Si quiere cambiar el nombre de un fichero cuando el MS-DOS lo copie en el disco de destino, simplemente añada el nuevo nombre a la orden COPY.

Por ejemplo:

```
A>COPY OPENACCT.DAT B:NEWACCT.DAT
        1 File(s) copied
```

Aquí no sólo se ha copiado OPENACCT.DAT de A: a B:, sino que también se le ha dado un nombre diferente (NEWACCT.DAT) en el disco B:.

Recuerde que puede usar caracteres comodín en los nombres de ficheros para copiar varios ficheros.

Por ejemplo,

```
A>COPY *.BAT B:
```

Esta orden copia todos los ficheros con la extensión BAT del disco A: al disco B:.

También se puede usar COPY para hacer un duplicado de un fichero en el mismo disco. Sin embargo, la copia tiene que tener un nombre diferente, puesto que el MS-DOS no permite dos ficheros con el mismo nombre en un disco.

Para hacer una copia del fichero JUDGES.TXT:

```
A>COPY JUDGES.TXT JUDGES.NEW
        1 File(s) copied
```

Como puede ver, el disco de destino no se ha especificado, así que la copia permanece en el mismo disco. Ahora usted tiene una nueva copia del fichero JUDGES.TXT (llamada JUDGES.NEW) en el mismo disco.

Si está copiando de otro disco al disco por defecto, sin cambiar el nombre del fichero, entonces no necesita incluir el nombre del disco de destino.

Por ejemplo, si A: es la unidad por defecto, y quiere copiar el fichero CASHFLOW.RPT de B: a A:, dé la orden

```
A>COPY B:CASHFLOW.RPT
```

La orden COPY moverá el fichero a la unidad por defecto.

También puede usar la orden COPY cuando quiera que el MS-DOS lea o escriba en los dispositivos hardware. Para hacerlo, use el nombre del dispositivo en vez de un nombre de fichero, y la orden COPY tratará los dispositivos como si fueran ficheros. Los nombres de los dispositivos se trataron en el capítulo 2. Generalmente, usted no usará dispositivos con la orden COPY, ya que usualmente es más fácil dejar que un programa de aplicación envíe los ficheros a los dispositivos.

Si define la puerta de comunicación (COM1:) como destino y un fichero OPERWX.FOR como origen, el MS-DOS pasará los datos del fichero al dispositivo:

```
A>COPY OPERWX.FOR COM1:
```

Esta orden envía el contenido de OPERWX.FOR a la puerta de comunicación.

Si define un dispositivo como origen, la orden COPY continuará leyendo del dispositivo hasta que usted introduzca un carácter CONTROL-Z seguido por un RETURN (el CONTROL-Z se llama *marca fin de fichero*). Este método de detectar el final de un fichero puede parecer un poco arbitrario; sin embargo, cuando use la orden COPY con un dispositivo, ésta continuará leyendo información del dispositivo hasta que vea una marca fin de fichero.

Si quiere escribir texto directamente en el fichero SHORT.MEM, usted usaría la siguiente orden:

```
A>COPY CON: SHORT.MEM
```

El nombre de dispositivo CON: representa al teclado y la pantalla. Ahora, todo lo que escriba se grabará directamente en el fichero. Usted observará que la petición de orden del MS-DOS no reaparece. Después de que haya introducido unas cuantas líneas, para acabar el fichero pulse CONTROL-Z y a continuación RETURN.

Cuando usted copia de un disquete maestro, tal como el original de su disco del sistema MS-DOS, puede que quiera ponerle una etiqueta de protección de escritura por si acaso da una orden COPY equivocada y copia información en el disquete.

Uso de COPY para combinar ficheros. La segunda sintaxis usual para la orden COPY es

COPY *origen1* + *origen2 destino*

Esta forma se usa cuando se quiere combinar ficheros, uno al final del otro, o, lo que es lo mismo, *conectarlos.* Se pueden concatenar tantos ficheros como se quiera en un fichero destino.

Por ejemplo, si quiere hacer un fichero llamado STAFF.TXT, que es una copia de JANE.TXT seguido por una copia de SYLVIA.TXT, dé la orden.

```
A>COPY JANE.TXT + SYLVIA.TXT STAFF.TXT
JANE.TXT
SYLVIA.TXT
        1 File(s) copied
```

La orden COPY lista cada fichero cuando lo añade al fichero destino, en este caso en el mismo disco.

Uso del argumento /V con COPY. Si el MS-DOS copia los datos incorrectamente, pueden ocurrir consecuencias desastrosas. Aunque el MS-DOS raramente hace una copia incorrecta, la mayoría de la gente cree que es mejor estar seguro. Por esta razón debería usar siempre el argumento /V después de cada orden COPY; o mejor todavía, siempre poner en ON (activar) el indicador VERIFY con la orden del MS-DOS VERIFY (descrita más adelante en este capítulo, en la sección «Indicadores del sistema»).

Aunque esto hace más lenta la orden, asegura que las copias son exactas. La seguridad de que el fichero se ha copiado correctamente, bien vale el tiempo extra que le lleva al MS-DOS el releer lo que ha escrito. El argumento /V se pone al final de la orden COPY.

Por ejemplo:

```
A>COPY PURCH.ACT B: /V
        1 File(s) copied
```

Esta orden asegura que el MS-DOS copia el fichero PURCH.ACT exactamente en el disco B:.

Otros usos

Otros dos argumentos, / A y / B se usan raramente con la orden COPY. Son bastante difíciles de comprender. Cuando se usan es para distinguir entre ficheros ASCII y *ficheros binarios* (de aquí la A y la B). Mientras un fichero ASCII contiene solamente texto normal, tal como una carta o una memoria, un fichero binario contiene caracteres especiales que no son de tipo texto. Los programas y las bases de datos son normalmente ficheros binarios. Los argumentos / A y / B se usan solamente cuando se copian ficheros que no se ajustan fácilmente a esas definiciones. Para más información sobre estos argumentos, consulte el manual de referencia del MS-DOS.

Reglas

No se puede copiar un fichero sobre sí mismo. Si intenta hacer una copia de un fichero en el mismo disco sin cambiar el nombre, el MS-DOS dará el mensaje «File cannot be copied onto itself» (Un fichero no puede copiarse sobre sí mismo).

Advertencias y errores frecuentes

El error más frecuente al usar la orden COPY es invertir el orden de los ficheros origen y destino. Esto es especialmente probable que le ocurra si está familiarizado con el sistema operativo CP/M, que requiere que el destino aparezca antes que el origen en la línea de la orden.

Si intenta copiar un fichero que no existe, el MS-DOS le dará un mensaje de error.

Por ejemplo, si intenta copiar el fichero TETHER.PRG y no existe:

```
A>COPY TETHER.PRG B:
TETHER.PRG File not found
        0 File(s) copied
```

El MS-DOS le dará el mensaje de error «File not found» (Fichero no encontrado).

Recuerde dar la opción / V, o poner VERIFY en ON, cuando copie ficheros. Esto evita los errores que pueden ocurrir si los ficheros se copian incorrectamente.

Antes de dar un dispositivo como origen en una orden COPY, asegúrese de que el dispositivo está debidamente preparado, sobre todo si la puerta de comunicaciones es el dispositivo origen. Recuerde que la orden COPY se que-

dará a la espera de los datos que se le envíen desde el dispositivo. Una vez que los datos comiencen a llegar, las orden COPY continuará leyendo del dispositivo hasta que vea el fin de fichero. Si, accidentalmente, usted da un nombre de dispositivo erróneo en una orden y no puede conseguir que el dispositivo responda, tendrá que hacer nuevamente la carga inicial de la computadora.

Sintaxis

La sintaxis de la orden COPY puede parecer extraña al principio pero, en realidad, es muy fácil de usar, porque los argumentos / A y / B apenas se usan. Las siguientes variaciones de la sintaxis se presentan en más de una línea, pero la orden COPY (igual que todas las órdenes del MS-DOS) se escribe siempre en una sola línea.

Copia de un fichero:

COPY [{/ A| / B}]*especifichero1* [{/ A| / B}]{*d:*| *especifichero2*} [{/ A| / B}] [/ V]

Concatenación de ficheros:

COPY [{/ A| / B}] *especifichero1a [*{/ A| / B}]
[+ *especifichero1b*] [{/ A| / B}] ... [/ V]
{*d:*| *especifichero2*} [{/ A| / B}] [/ V]

Copia en o de un dispositivo:

COPY {*especifichero1*|*dispositivo1*} {*especifichero2*|*dispositivo2*}

En cada tipo de sintaxis, *especifichero1* es el origen de la copia y *especifichero2* es el destino. La orden COPY siempre copia del primer argumento al segundo. También se pueden usar caracteres comodín en los nombres de los ficheros para copiar un grupo de ficheros.

{/A| /B} indica que se puede usar o bien / A o bien / B con cualquier especificador de fichero. El significado de / A y / B depende de si se utilizan en ficheros origen o en ficheros destino. / A y / B pueden aparecer antes o después del *especifichero* origen.

/ V hace que la orden COPY verifique que los datos escritos en el disco de destino son exactamente los mismos que había en el disco origen.

El argumento *dispositivo* representa a cualquier nombre de dispositivo en el MS-DOS, tal como LPT1: para la impresora, COM1: para la puerta de comunicaciones, etc. Estos dispositivos y sus nombres en el MS-DOS se tratan en el capítulo 2.

Orden COMP
Externa
Sólo IBM

La orden COMP compara los contenidos de dos ficheros e informa de las diferencias en la pantalla. Aunque esta orden es específica del PC-DOS de IBM, algunos otros fabricantes de hardware tiene órdenes similares.

Usos más comunes

La orden COMP se puede usar para comparar dos ficheros. La sintaxis usual es

COMP *especifichero1 especifichero2*

El comprobar la exactitud de un fichero, que se acaba de copiar, con la orden COMP da la tranquilidad de que la copia se ha realizado con éxito, aunque la opción /V de la orden COPY es suficiente. Puede que usted desee usar la orden COMP después de hacer una copia de un fichero.

Por ejemplo:

```
A>COPY INVOICES.DAT B:
        1 File(s) copied

A>COMP A:INVOICES.DAT B:

A:INVOICES.DAT and B:INVOICES.DAT

Files compare ok

Compare more files (Y/N)?N
```

Observe que no necesita repetir el nombre del segundo fichero si es idéntico al del primero. El primer mensaje (Files compare ok) le indica que ambos ficheros son iguales. El segundo mensaje le pregunta si quiere comparar más ficheros. La respuesta debe ser Y (sí) o N (no).

La versión 2 del MS-DOS permite que la orden COMP se use para comprobar todos los ficheros de un disco, diciendo simplemente que compare *.* de un disco con otro.

Por ejemplo:

```
A>COMP *.* B:
```

Entre cada conjunto de ficheros, la orden COMP pregunta

```
Compare more files (Y/N)?
```

Conteste Y si quiere seguir comparando ficheros.

La orden COMP no es recomendable para comparar ficheros si usted sabe que uno de ellos ha sido modificado. No se pueden comparar dos ficheros de diferentes tamaños, aunque el tamaño sólo difiera en un carácter. Si usted sólo ha reemplazado caracteres (sin alterar el tamaño), la respuesta de la orden COMP será muy difícil de leer. Esto ocurre porque todas las diferencias se muestran con números hexadecimales, no con letras.

Para ver por qué no debería usar COMP para comparar ficheros diferentes suponga que usted compara un fichero llamado OPERA.DOC con una copia de seguridad algo diferente, llamada OPERA.BAK. En su pantalla saldría algo parecido a lo siguiente:

```
A>COMP OPERA.DOC OPERA.BAK

A:OPERA   .DOC and A:OPERA   .BAK

Compare error at offset 4D
File 1 = 58
File 2 = 59

Compare error at offset 51
File 1 = 58
File 2 = 59

Compare error at offset 6A
File 1 = 53
File 2 = 56

    .
    .
    .
```

Como puede ver, la orden COMP le permite saber que hay diferencias, pero no hace fácil el descifrar en qué consisten esas diferencias.

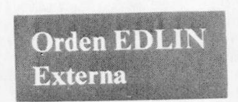
Para comparar dos disquetes, la orden DISKCOMP es mucho más rápida que la comparación fichero a fichero. Además comprueba las pistas de carga inicial y las áreas de directorio. La orden DISKCOMP se trata más adelante en este capítulo, en la sección «Ordenes de mantenimiento de discos».

Sintaxis

La sintaxis de la orden COMP es

COMP [*especifichero1 especifichero2*]

Los argumentos *especifichero1* y *especifichero2* son los ficheros que se van a comparar. Se permiten las especificaciones con caracteres comodín. Si se olvida de dar los nombres de los ficheros, o si da sólo uno cuando se requieren dos, la orden COMP le pedirá que los introduzca.



La orden EDLIN es un editor de líneas que se puede usar para crear y modificar el contenido de los ficheros texto.

Usos más comunes

La forma más corriente de la orden EDLIN es

EDLIN *especifichero*

EDLIN es un editor de líneas, lo que significa que sólo se puede trabajar con una línea del fichero a la vez. Este editor no aprovecha las posibilidades de la pantalla completa de la computadora. EDLIN no es recomendable porque es difícil de usar y poco potente.

Este libro no describe el modo de operación de EDLIN. Para hacerlo, harían falta muchas más páginas de las disponibles. EDLIN no es fácil de aprender a usar y, a menos que se conozca bien, su uso puede ser contraproducente. Si usted realmente quiere usar EDLIN, vea el manual de su computadora o uno de los libros que se citan en el capítulo 7.

Si el fabricante de su computadora, no le proporcionó un programa editor de texto junto con la máquina, debería mirar entre los muchos editores de texto disponibles por menos de 100 $; algunos incluso son gratuitos. Las asociaciones de usuarios frecuentemente regalan copias de programas editores de texto. El capítulo 6 habla de muchos editores y procesadores de texto y de lo

que pueden hacer por usted; estos programas son mucho más fáciles de usar que EDLIN.

Sintaxis

La sintaxis de la orden EDLIN es

EDLIN *especifichero* [/B]

El argumento *especifichero* es el nombre del fichero que se va a editar. Si se incluye el argumento /B, EDLIN tratará el fichero como fichero binario y leerá todos los datos de cada sector.

ORDENES DE SALIDA DE FICHEROS

En esta sección se tratan las siguientes órdenes:

TYPE	*Visualiza un fichero en la pantalla*
PRINT	*Imprime un fichero en la impresora*
MORE	*Visualiza un fichero en la pantalla con pausas*

La sección previa trataba de cómo mover ficheros, de cómo eliminarlos y de cómo darles nuevos nombres. Esta sección describre otra cosa importante que usted quiere hacer con los ficheros: examinarlos. Las órdenes TIPE, PRINT y MORE se usan para visualizar los ficheros en la pantalla o imprimirlos en un papel.

Estas órdenes son *órdenes secuenciales.* Esto quiere decir que, cuando use una de estas órdenes, sólo puede examinar un fichero desde el principio al fin; no puede volver para atrás y releer un trozo de fichero. En este sentido las órdenes son restrictivas, puesto que la gente a menudo quiere dar vueltas dentro del fichero, mientras lee, y reexaminar la información.

Para tener más flexibilidad, puede usar un editor o procesador de texto para visualizar el contenido de un fichero. Casi todos los editores de texto le permiten ver un fichero texto y moverse hacia adelante y hacia atrás, por todo el fichero. Así, se debería usar un procesador de texto para releer ficheros, excepto cuando los ficheros son cortos o cuando se quiere guardar una copia impresa del fichero.

No se pueden visualizar programas u otros ficheros binarios con las órdenes discutidas en esta sección. Estos ficheros que no se pueden visualizar tienen normalmente las extensiones COM, EXE, OVR y BIN. Los ficheros no visualizables pueden tener hasta 256 tipos de caracteres posibles, que pueden aparecer en la pantalla como caracteres gráficos, o puede que no aparezcan en

absoluto. En cualquier caso, no se puede utilizar la información de estos ficheros simplemente leyéndola. Si realmente quiere ver los contenidos de estos ficheros, puede usar la orden DEBUG tratada en la sección de este capítulo «Herramientas de programación».

Orden TYPE
Interna

La orden TYPE visualiza un fichero texto en la pantalla.

Usos más comunes

La orden es muy fácil de usar. Simplemente dé la orden TYPE seguida por el nombre del fichero:

TYPE *especifichero*

El MS-DOS escribirá entonces el fichero en la pantalla.

Por ejemplo, para ver el fichero LEADSRPT.TXT en la pantalla, escriba la orden

```
A>TYPE LEADSRPT.TXT
                              Sales Report

                              Tim Meyerstein


New leads generated:
     .
     .
     .
```

La orden TYPE hace un *desplazamiento* (scroll) sobre el fichero (es decir, muestra el fichero en la pantalla, añadiendo cada vez una nueva línea). Por esta razón, si el fichero tiene más de 24 ó 25 líneas, qué es lo que coge en la pantalla, probablemente ocurrirá que el texto se moverá tan deprisa que será imposible leerlo. Para leer el fichero, puede usar una tecla especial (o una

combinación de teclas) que le permite congelar el texto en la pantalla. Por ejemplo, en el IBM PC, la combinación de teclas es CONTROL-NUMLOCK. Para parar la orden TYPE, pulse CONTROL-C y volverá al MS-DOS.

Si el fichero que está escribiendo tiene tabuladores (caracteres generados al pulsar la tecla TAB), la orden TYPE saltará hasta la siguiente columna que sea múltiplo de 8 (es decir, la columna 8, 16, 24, etc.) e imprimirá el texto que siga al carácter tabulador. Así, en la impresión del último ejemplo, los primeros caracteres impresos aparecen en la columna 24 porque había 3 caracteres tabuladores al comienzo de la línea.

Advertencias y errores frecuentes

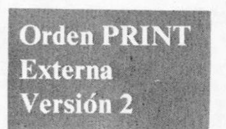
La orden TYPE no impide que usted escriba ficheros de programas y ficheros binarios en la pantalla. Sin embargo, si usa TYPE en estos ficheros, probablemente usted verá un galimatías.

Sintaxis

La sintaxis de la orden TYPE es

TYPE *especifichero*

El argumento *especifichero* es el fichero que se va a visualizar. No se pueden usar caracteres comodín en la especificación del fichero.



Orden PRINT
Externa
Versión 2

La orden PRINT imprime ficheros en la impresora y le permite continuar trabajando con el MS-DOS al mismo tiempo. Permite crear una cola de ficheros para que se impriman uno detrás de otro.

Usos más comunes

La orden PRINT es un método sofisticado de imprimir ficheros por la impresora. Otros sistemas operativos permiten enviar sólo un fichero cada vez a la impresora, a menudo obligando a esperar a que se termine la impresión del

fichero antes de que se puedan dar nuevas órdenes. La sintaxis normal de la orden PRINT es

PRINT *especifichero*

o bien

PRINT *especifichero1 especifichero2 ...*

Se pueden meter hasta 10 ficheros de una vez en una cola de espera para impresión.

La primera vez que ejecute la orden PRINT después de la carga inicial del MS-DOS, se le preguntará el nombre del dispositivo de listado. Este es el nombre del dispositivo al cual se enviarán los ficheros para imprimir. Generalmente, el nombre del dispositivo es PRN o LPT (los nombres de la puerta de tipo paralelo de impresión). Cuando dé el nombre del dispositivo asegúrese de *no* incluir los dos puntos al final. Si tiene una impresora de tipo serie, el dispositivo se llamará probablemente COM1 o COM2.

Si simplemente pulsa RETURN, cuando se le hace la pregunta, la orden PRINT asumirá que usted quiere PRN.

Por ejemplo, si quiere imprimir el fichero HOSPT12.RPT y es el primer fichero que va a imprimir desde que hizo la carga inicial de la computadora, escriba la orden

```
A>PRINT HOSPT12.RPT
Name of list device [PRN]:COM1
Resident part of PRINT installed.

          A:HOSPT12 .RPT is currently being printed
```

La orden PRINT imprimiría entonces el fichero en el dispositivo especificado como salida (en este caso, COM1:, la puerta serie).

Antes de dar la orden PRINT debería estar seguro de que el dispositivo de salida está conectado y encendido. Desafortunadamente, la orden PRINT no le dirá cuando hay algún problema en el dispositivo de salida pero, la próxima vez que intente dar una orden de imprimir, usted lo sabrá porque el dispositivo no querrá imprimir.

Por ejemplo, si la impresora no está encendida, usted verá algo parecido a esto:

```
A>PRINT ACCTSRPT.TXT

          A:ACCTSRPT.TXT is currently being printed
```

```
A>PRINT MYFILE
Errors on list device indicate that it
may be off-line. Please check it.

          A:ACCTSRPT.TXT is currently being printed
          A:MYFILE   .   is in queue
```

El mensaje indica que la impresora puede estar «off-line» (esto es un modo de decir que «no está trabajando como se espera»). El resultado es que ninguno de los dos ficheros se imprime.

Introducción de ficheros en una cola de espera para impresión. Si quiere imprimir un conjunto de ficheros, uno tras otro, puede dar hasta 10 nombres después de la orden PRINT. Los caracteres comodín también se pueden utilizar para formar una cola de ficheros. Para ver qué ficheros están en la cola de espera, escriba la orden PRINT sin argumentos.

Por ejemplo, para imprimir el fichero NEWPROG.BAS y todos los ficheros que tienen la extensión COB, introduzca la orden

```
A>PRINT NEWPROG.BAS *.COB
A>PRINT

          A:NEWPROG .BAS is currently being printed
          A:TSTAMP  .COB is in queue
          A:TSTAMP1 .COB is in queue
          A:TSTAMP2 .COB is in queue
```

Los ficheros se imprimen ahora por la impresora sin que usted tenga que introducir nada más.

Utilización de los argumentos de la orden PRINT. Si usted da una orden PRINT pero, a continuación, decide que no quiere imprimir uno de los ficheros que está en la cola, puede usar el argumento /C de la orden PRINT para cancelar un fichero particular. Simplemente escriba la orden PRINT seguida por el nombre de ese fichero y después /C.

Por ejemplo, para cancelar el fichero TSTAMP1.COB, introduzca la orden

```
A>PRINT TSTAMP1.COB/C

          A:NEWPROG .BAS is currently being printed
          A:TSTAMP  .COB is in queue
          A:TSTAMP2 .COB is in queue
```

Otro argumento es el argumento /T. Puede usarlo para cancelar todos los ficheros de la cola.

Por ejemplo, escriba la orden

```
A>PRINT /T

PRINT queue is empty.
```

Todos los ficheros se han eliminado de la cola de espera, con lo cual ahora está vacía.

Cuando cancele ficheros con los argumentos /C o /T, la orden PRINT hará que suene la campanilla en la impresora, escribirá un mensaje por impresora y avanzará el papel hasta el comienzo de la siguiente página.

La orden PRINT expande los caracteres tabuladores, de la misma manera que la orden TYPE.

Otros usos

Se puede usar el argumento /P en combinación con el argumento /C o con /T para introducir en la cola nuevos ficheros. El argumento /P permite dar una sola orden que cancela ficheros de la cola y permite añadir más. No se usa frecuentemente.

Reglas

El disco que contiene el fichero que se quiere imprimir, tiene que mantenerse en la unidad de disco adecuada, hasta que el fichero se haya terminado de imprimir. Si usted lo saca, se imprimirá un mensaje por impresora, y ésta se parará.

No modifique ni borre un fichero que esté en la cola de espera antes de que se termine de imprimir.

Puesto que el programa PRINT permanece en memoria después de que usted lo pone en funcionamiento, su RAM disponible disminuye en aproximadamente 3.000 bytes (3K).

Advertencias y errores frecuentes

El error más frecuente es no tener la impresora encendida cuando se da la orden PRINT. Afortunadamente, la orden PRINT espera hasta que la impresora esté encendida antes de intentar imprimir los ficheros.

Sintaxis

La sintaxis de la orden PRINT es

PRINT [*especifichero* ... [/T] [/C] [/P] ...]

El argumento *especifichero* es el fichero que se va a imprimir. Se pueden especificar muchos ficheros para imprimir, bien con caracteres comodín o dando una lista de nombres de los ficheros individuales. Si no se dan argumentos, la orden PRINT escribirá la lista de los ficheros que están en la cola de espera.

El uso del argumento /T cancela la cola de ficheros. El argumento /C permite que se cancelen ficheros de la cola. /C cancela todos los ficheros de la cola hasta que se dé el argumento /P. Generalmente, el argumento /P sólo se usa si se ha dado un argumento /T o /C en la misma orden.

Orden MORE
Externa
Versión 2

La orden MORE escribe un fichero en el monitor, ocupando una pantalla entera cada vez y esperando a que se pulse alguna tecla antes de continuar. La orden MORE utiliza entrada redirigida, que se describe en el capítulo 4, pero usted no necesita saber cómo usar la entrada redirigida para usar la orden MORE.

Usos más comunes

La orden MORE es más fácil de usar que la orden TYPE para ficheros de más de 20 líneas. La sintaxis usual es

MORE < *especifichero*

El signo menor que (<) le dice al MS-DOS que usted está usando entrada redirigida.

Después de visualizar 23 líneas de su fichero (si tiene una pantalla de 25 líneas), el programa escribe

```
-- More --
```

A continuación, espera a que usted pulse alguna tecla antes de continuar. Cuando pulsa una tecla se visualiza la siguiente pantalla. La orden MORE expande los tabuladores de la misma forma que la orden TYPE.

Por ejemplo, si escribe la orden

```
A>MORE <DATACOMM.RPT
```

el fichero DATACOMM.RPT se visualizará pantalla a pantalla.

Advertencias y errores frecuentes

Si olvida poner el < antes del nombre del fichero, la orden MORE simplemente repetirá cada línea que usted escriba. Puesto que claramente no es esto lo que usted quiere, pulse CONTROL-C para volver al MS-DOS.

Sintaxis

La sintaxis para la orden MORE es

MORE < *especifichero*

El signo menor que (<) es necesario para que MORE funcione. El argumento *especifichero* es el fichero que se va a visualizar.

ORDENES DE MANTENIMIENTO DE DISCOS

Esta sección trata las siguientes órdenes:

DIR	*Visualiza la lista de los ficheros de un disco*
FORMAT	*Prepara un disco para el MS-DOS*
SYS	*Graba los ficheros del sistema y las pistas de carga inicial en un disquete*
CHKDSK	*Examina y repara discos*
RECOVER	*Repara ficheros con sectores en mal estado*
DISKCOPY	*Copia un disquete entero*
DISKCOMP	*Compara dos disquetes*
FDISK	*Realiza el mantenimiento del IBM PC/XT*
BACKUP	*Hace copias de seguridad del disco duro*
RESTORE	*Hace copias de ficheros de seguridad*

Esta sección cubre las órdenes que tienen que ver con los discos y con los directorios. Puede usar la orden DIR para averiguar si ciertos ficheros están

en un disco particular o para saber la longitud de un fichero. FORMAT, DISKCOPY y SYS se utilizan para preparar los disquetes flexibles para su uso. Después de utilizar la orden DISKCOPY, la orden DISKCOMP permite comparar los discos para determinar si la copia es exacta.

Nadie planea que los discos flexibles o duros se vuelvan ilegibles. Sin embargo, a pesar de sus buenas intenciones, puede que a veces el MS-DOS no pueda leer el disco por alguna razón. Si encuentra errores en los discos flexibles o duros, debería usar las órdenes CHKDSK y RECOVER. Estas órdenes ayudan a reparar discos ilegibles: CHKDSK repara los errores del directorio del disco y RECOVER repara los errores físicos del disco (también conocidos como *sectores en mal estado*). Estas dos órdenes le ayudarán a determinar lo que está mal, e incluso a arreglarlo. Si no funcionan lea cómo solucionar incidentes en el capítulo 5.

Orden DIR
Interna

La orden DIR lista los ficheros de un disco y da información de sus tamaños y de cuando han sido actualizados por última vez.

Usos más comunes

El uso más corriente de la orden DIR es litar todos los ficheros que están en la unidad de disco por defecto. La sintaxis usual es

DIR [*d:*]

Por ejemplo, para introducir la orden DIR, escriba

```
A>DIR

 Volume in drive A is SALES
 Directory of   A:\

CALC      EXE     25600    5-20-84   12:58p
DRIVER    EXE     32420    1-27-84   10:35a
SALES84   RPT      7168    8-03-84    8:04p
MAKEFILE  BAT      1664    4-01-84   11:10p
TOTSALES  DAT      5888    3-08-84   12:09p
MARTY     DAT      1280    3-08-84   12:12p
RDBMS     COM     39936    4-27-84   12:19p
RDBMS     OVL     33120    4-27-84   12:19p
         8 File(s)     11264 bytes free
```

El MS-DOS visualiza una lista de ficheros semejante a la mostrada aquí.

El listado dado por DIR contiene varios elementos que puede que usted no reconozca:

- La información en la parte de arriba del listado, «Volume in drive A is SALES» («El volumen que está en la unidad A es SALES»), da el *rótulo del disco* (y el nombre del camino actual, que se describe en el capítulo 4). El rótulo lo pone usted cuando usa la orden FORMAT con la opción /V. Si usted no rotuló su disco, en esta línea aparecería «Volume in drive A has no label» («El volumen que está en la unidad A no tiene rótulo»). Si está usando la versión 1 del MS-DOS no verá esta información inicial.
- Las dos primeras columnas del listado contienen el nombre del fichero y la extensión. En el listado, estas dos partes del especificador del fichero no están separadas por un punto.
- Si su disco tiene un sistema de subdirectorios (que es una característica, disponible en la versión 2 del MS-DOS, que permite tener muchos directorios en un disco), habrá entradas para ficheros llamados . y .. (a no ser que el subdirectorio actual sea la raíz). El capítulo 4 explica los subdirectorios en detalle.
- La tercera columna contiene el tamaño del fichero en bytes. (Si usted está usando caminos, en esta columna puede que aparezca <DIR>, que no es un fichero, sino el nombre de un subdirectorio).
- Las últimas dos columnas dan la fecha y la hora de la actualización más reciente del fichero. Si el fichero no se ha actualizado nunca, estas columnas dan la fecha y la hora en que se creó. (Si su computadora no tiene un calendario de tiempo real y usted no introduce rutinariamente la fecha y la hora cuando hace la carga inicial, estas fechas y horas pueden ser incorrectas).
- La última línea indica cuántos ficheros están en el listado y cuántos bytes quedan libres en el disco para otros ficheros. Si está usando la versión 1 del MS-DOS, esta línea le dirá el número de ficheros, pero no la cantidad de espacio libre.

También se puede ver el directorio de otro disco especificando dicho disco.

Por ejemplo,

```
A>DIR B:
```

Esta orden hace que el MS-DOS liste el directorio del disco B: en la pantalla.

Uso de DIR con ficheros individuales. La orden DIR permite examinar ficheros individuales. También se puede conseguir un listado de un grupo de ficheros usando caracteres comodín en el nombre del fichero.

Por ejemplo, suponga que quiere ver la entrada correspondiente a MARTY.DAT. Daría la orden

```
A>DIR MARTY.DAT

 Volume in drive A is SALES
 Directory of A:\

MARTY     DAT      1280    3-08-84   12:12p
        1 File(s)       11264 bytes free
```

Si quiere ver todos los ficheros con extensión DAT, dé la orden

```
A>DIR *.DAT

 Volume in drive A is SALES
 Directory of  A:\

TOTSALES DAT      5888    3-08-84   12:09p
MARTY    DAT      1280    3-08-84   12:12p
        2 File(s)       11264 bytes free
```

El uso del carácter comodín (*) en el nombre del fichero o en la extensión es tan común que la orden DIR le permite omitirlo.

Por ejemplo, la orden previa (DIR *.DAT) podría darse también como

```
A>DIR .DAT
```

El carácter comodín es innecesario. Sin embargo, el punto que precede a la extensión es necesario, puesto que le dice al MS-DOS que DAT es una extensión.

Modificación del listado de DIR con argumentos. La orden DIR puede usarse con dos argumentos, /W y /P. Si el directorio tiene un gran número de ficheros y no tiene interés en la información del tamaño ni en la fecha de actualización, puede usar el argumento /W para simplificar el listado.

Por ejemplo, observe el cambio entre los dos siguientes directorios:

```
A>DIR C:

 Volume in drive C has no label
 Directory of  C:\

COMMAND  COM     17664    3-08-83   12:00p
ANSI     SYS      1664    3-08-83   12:00p
FORMAT   COM      6016    3-08-83   12:00p
CHKDSK   COM      6400    3-08-83   12:00p
SYS      COM      1408    3-08-83   12:00p
DISKCOPY COM      2444    3-08-83   12:00p
```

```
DISKCOMP COM     2074   3-08-83   12:00p
COMP     COM     2523   3-08-83   12:00p
EDLIN    COM     4608   3-08-83   12:00p
MODE     COM     3139   3-08-83   12:00p
FDISK    COM     6177   3-08-83   12:00p
BACKUP   COM     3687   3-08-83   12:00p
RESTORE  COM     4003   3-08-83   12:00p
PRINT    COM     4608   3-08-83   12:00p
RECOVER  COM     2304   3-08-83   12:00p
ASSIGN   COM      896   3-08-83   12:00p
TREE     COM     1513   3-08-83   12:00p
GRAPHICS COM      789   3-08-83   12:00p
SORT     EXE     1280   3-08-83   12:00p
FIND     EXE     5888   3-08-83   12:00p
MORE     COM      384   3-08-83   12:00p
BASIC    COM    16256   3-08-83   12:00p
BASICA   COM    25984   3-08-83   12:00p
        23 File(s)      31232 bytes free

A>DIR C: /W

 Volume in drive C has no label
 Directory of  C:\

COMMAND  COM   ANSI     SYS   FORMAT   COM   CHKDSK   COM   SYS      COM
DISKCOPY COM   DISKCOMP COM   COMP     COM   EDLIN    COM   MODE     COM
FDISK    COM   BACKUP   COM   RESTORE  COM   PRINT    COM   RECOVER  COM
ASSIGN   COM   TREE     COM   GRAPHICS COM   SORT     EXE   FIND     EXE
MORE     COM   BASIC    COM   BASICA   COM
        23 File(s)      31232 bytes free
```

El argumento / P hará que el listado del directorio haga una pausa al final de cada página. En la parte de abajo de la pantalla aparecerá el mensaje «Strike a Key when ready»...» («Pulse una tecla cuando esté preparado...»). Esto es similar a la petición «—More—» que se visualizaba en la orden MORE. La orden DIR esperará a que se pulse una tecla antes de continuar.

Uso de DIR para subdirectorios. Si está usando caminos, puede usar la orden DIR para obtener el directorio de un subdirectorio, simplemente nombrándolo en la orden.

Por ejemplo:

```
A>DIR

 Volume in drive A has no label
 Directory of  A:\data

.             <DIR>      1-19-84    6:20a
..            <DIR>      1-19-84    6:20a
SHELLEY  DAT     17664   3-08-83   12:00p
TONY     DAT      1664   3-08-83   12:00p
PHONES        <DIR>      2-18-84   10:55p
        5 File(s)    4631232 bytes free

A>DIR PHONES

 Volume in drive A has no label
 Directory of  A:\data\phones
```

```
.            <DIR>        1-19-84    6:20a
..           <DIR>        1-19-84    6:20a
PHONES    DAT     9191    3-08-83   12:00p
        3 File(s)    4631232 bytes free
```

Observe que la segunda línea le muestra el subdirectorio. Puede listar el directorio del subdirectorio padre dando la orden

```
A>DIR ..
```

Para más información relativa a caminos y subdirectorios, vea más adelante en este capítulo «Ordenes de mantenimiento de caminos» y también el capítulo 4.

Reglas

En el listado dado por DIR no aparecen todos los ficheros. Algunos ficheros están ocultos para la orden DIR. Estos ficheros tienen una marca especial en su entrada del directorio que le indica a la orden DIR que no debe listarlos (también se llaman ficheros «secretos»). Usted no puede crear estos ficheros; se crean con la orden FORMAT y con la orden SYS. Usted debería conocerlos, puesto que se listarán en la salida de la orden CHKDSK.

Sintaxis

La sintaxis de la orden DIR es

DIR [*d:*| *especifichero*] [/P] [/W]

Si usted da la orden DIR sin argumentos, listará el directorio del disco por defecto. Si da un *d:,* visualizará el directorio de ese disco; si da un *especifichero* (que puede contener caracteres comodín), sólo se listarán los ficheros que tengan ese *especifichero.*

El argumento /W visualiza los ficheros en cinco columnas, de un lado a otro de la pantalla, pero no muestra el tamaño del fichero ni la última actualización. El argumento /P hace que el listado haga una pausa al final de cada página.

Orden FORMAT
Externa

La orden FORMAT prepara un disco, flexible o duro, para su uso con el MS-DOS. Algunos fabricantes de discos duros requieren el uso de una orden diferente para formatear sus discos. En el IBM PC, la orden FORMAT puede preparar los disquetes de forma que se puedan leer por todas las versiones del PC-DOS.

Usos más comunes

Cuando se compran los disquetes, generalmente no están formateados; hay que usar la orden FORMAT antes de intentar grabar ficheros en un disquete. (El capítulo 2 describió algunos de los conceptos que están detrás del formateado de discos). Puesto que el formateado de un disco borra toda la información del disco, tiene que usar esta orden con precaución. Use la orden FORMAT para preparar discos para su uso con el MS-DOS.

La sintaxis usual es

FORMAT [*d:*]

Para formatear un disco en la unidad B:, escriba la orden

```
A>FORMAT B:
```

El programa FORMAT responde

```
Insert new diskette for drive B:
and strike any key when ready
```

(«Inserte un disquete nuevo en la unidad B: y pulse cualquier tecla cuando esté preparado»). Después de pulsar una tecla, el programa formatea el disco y escribe

```
Formatting...
Format complete.
Format another (Y/N) ?
```

Si quiere formatear de nuevo la misma unidad, pulse Y.

Uso de FORMAT con argumentos. La orden FORMAT se puede usar con varios argumentos. El argumento /S le dice a la orden FORMAT que escriba las pistas de carga inicial y los ficheros del sistema en el disquete, después de formatearlo. Esto hace que el disquete sea un disco del sistema, que quiere decir que usted puede hacer la carga inicial del sistema con él. La sintaxis usual de la orden FORMAT con el argumento /S es

FORMAT [*d:*] /S

No es necesario usar el argumento /S en un disco, a no ser que se quiera usar como disco del sistema. Esto es porque el MS-DOS lee del disquete las pistas del sistema sólo cuando se hace la carga inicial de la computadora. (Esto es diferente de otros sistemas operativos, tales como el CP/M, que lee las pistas cada vez que se sale de un programa). Puesto que los ficheros del sistema ocupan 39K, se puede perder una buena cantidad de espacio poniendo el sistema operativo en un disco donde no hace falta.

Por ejemplo, después de formatear el disco con el argumento /S, el programa FORMAT escribirá

```
System transferred

   179712 bytes total disk space
    29936 bytes used by system
   139776 bytes available on disk
```

Esto muestra que parte del espacio del disco lo ocupan los ficheros del sistema.

El argumento /S no copia el fichero COMMAND.COM en el disco formateado. Este fichero no es un fichero del sistema, pero es necesario para ejecutar el MS-DOS. Así, si usted quiere hacer un disco que se va a usar como disquete para la carga inicial, tiene que copiar también COMMAND.COM en el disco.

Por ejemplo:

```
A>FORMAT B: /S

A>COPY COMMAND.COM B:
```

El resultado es un disquete que se puede usar en la carga inicial.

Naturalmente, usted puede copiar todos los ficheros del sistema en vez de copiar solamente COMMAND.COM. El resultado será un disco del sistema.

El argumento /V (con la versión 2 del MS-DOS) le permite poner un *rótulo de volumen* (también conocido como rótulo del disco) al disquete. El rótulo del volumen no es lo mismo que la etiqueta de papel de la parte de fuera del disquete; es un dato especial de 11 caracteres grabado en el directorio del disquete. Algunos programas de aplicación usan estos rótulos para identificar el contenido de un disco. Sin embargo, los rótulos de volumen son poco usados por la mayoría de la gente.

La orden FORMAT es la única orden del MS-DOS que le permite escribir un rótulo en un disco y sólo se puede ver este rótulo con las órdenes DIR, VOL y CHKDSK.

Por ejemplo, si usa el argumento /V, el programa le pedirá el rótulo después de formatear el disco:

```
Volume label (11 characters, ENTER for none)?
```

Escriba el rótulo del volumen con los caracteres válidos para nombres de ficheros. Pulse ENTER si no quiere dar un rótulo de volumen.

Especificación de pistas y sectores para el IBM PC. Si usted tiene unidades de disquetes de doble cara y quiere hacer disquetes que alguien que tenga unidades de una sola cara pueda leer, tiene que usar el argumento /1 en la orden FORMAT. Esto es así porque FORMAT normalmente formatearía un disquete para unidades de doble cara (si usted tiene estas unidades).

Si está usando la versión 2 del MS-DOS, y quiere hacer disquetes que se puedan leer con la versión 1, tiene que usar el argumento /8 en la orden FORMAT. La razón es que FORMAT normalmente formatea un disquete con 9 sectores por pista en la versión 2, mientras que las computadoras que usan la versión 1 sólo pueden leer discos formateados con 8 sectores por pista.

Por ejemplo, si está usando la versión 2 y quiere formatear un disquete que se pueda usar con unidades de una sola cara bajo la versión 1 del MS-DOS, dé la orden

```
A>FORMAT /1/8
```

El MS-DOS le pedirá que inserte el disquete que se va a formatear en la unidad A: y, a continuación, lo formateará con una sola cara y 8 sectores por pista.

El argumento /B se utiliza para formatear un disquete con 8 sectores por pista y espacio reservado para los ficheros ocultos del sistema. Este argumento hace un disquete legible con las versiones 1 y 2 del MS-DOS. Sin embargo, no graba los ficheros en el disquete. Usted tiene que escribir los ficheros del sistema con la orden SYS, descrita en la próxima sección de este capítulo.

Por esta razón, después de usar /B, usted puede darle el disquete formateado a otra persona sin darle realmente las pistas y ficheros del sistema (que es lo que no está permitido con la licencia de Microsoft). La persona que recibe el disquete puede usar a continuación la orden SYS para hacer copias de los ficheros del sistema en el disquete.

Advertencias y errores frecuentes

Recuerde que la orden FORMAT borra toda la información del disco. Por esta razón, cuando se formatea un disco que ya ha tenido información, hay que estar seguro de que esa información ya no se necesita.

Si quiere que el disquete formateado sea eventualmente un disquete del sistema, tiene que formatearlo con el argumento /S o bien con el argumento /B. Esto es debido a la forma en que las pistas de carga inicial encuentran a los ficheros del sistema.

Si las tres primeras entradas en el directorio del disquete no son los ficheros del sistema, el disquete no funcionará como disquete de carga inicial. La

orden SYS (que escribe en un disquete las pistas de carga inicial y los ficheros del sistema) sólo funcionará si el disquete se ha formateado con el argumento /S o /B, o bien si el disquete no ha sido grabado desde que se formateó.

Sintaxis

La sintaxis para la orden FORMAT es

FORMAT [*d*:] [/S] [/1] [/8] [/V] [/B]

El orden de los argumentos no es importante. *d:* es la unidad de disco que se va a formatear. Si no se especifica la unidad, la orden FORMAT asume que se quiere formatear la unidad por defecto.

El argumento /S hace que las pistas de carga inicial y los ficheros del sistema se escriban en el disquete formateado. El argumento /1 indica que se quiere formatear el disquete para unidades de una sola cara. El argumento /8 indica que se quiere formatear el disquete con 8 sectores por pista. El argumento /V pone un rótulo de volumen al disquete. El argumento /B se usa para formatear un disquete con 8 sectores por pista y espacio reservado para los ficheros ocultos del sistema.

Orden SYS
Externa

La orden SYS copia las pistas de inicialización y los ficheros del sistema de un disquete a otro. Sin embargo, el disco no funcionará como disquete de carga inicial si el segundo disquete no está correctamente formateado antes de ejecutar la orden SYS.

Usos más comunes

El empleo más frecuente de la orden SYS es el de hacer que los discos protegidos contra copia, puedan realizar la carga inicial del sistema. (Estos disquetes son usualmente el software de aplicación que se compra). Usted hace esto copiando su sistema operativo en el disquete del programa de aplicación. La razón de por qué los disquetes de programas de aplicación vienen sin el sistema operativo es que es ilegal vender o dar un disquete con una copia del sistema operativo sin el consentimiento del fabricante. Sin embargo, a usted le está permitido hacer copias del MS-DOS siempre que las ejecute solamente en su computadora. La sintaxis general de la orden es

SYS [*d*:]

Si el disquete no está protegido contra copia, usted podría formatear un disquete en blanco con la orden FORMAT y el argumento /S, copiar el fichero COMMAND.COM de su disquete del sistema al nuevo disquete, y, a continuación, copiar los ficheros del disquete de aplicación al nuevo disquete. En este caso, no necesitaría usar la orden SYS.

Antes de usar la orden SYS, los discos tienen que estar formateados, como se discutió en la sección precedente sobre la orden FORMAT.

Para añadir las pistas de carga inicial y los ficheros del sistema al disquete de la unidad B:, use la orden

```
A>SYS B:
```

Reglas

El disco sobre el que se escribe tiene que haber sido formateado con el argumento /S o el argumento /B de la orden FORMAT o bien no haber sido usado desde que se formateó.

Advertencias y errores frecuentes

Si usted formatea un disco sin el argumento /S ni el /B y a continuación escribe algo en él, ya no puede usar la orden SYS para convertir el disco en un disco que pueda realizar la carga inicial, ni siquiera aunque borre el fichero. Si usa la orden SYS sobre un disquete así, puede perder datos del disco. Por esta razón, asegúrese de que no hay ficheros en el disco antes de dar la orden SYS.

La orden SYS no copia el fichero COMMAND.COM, que es necesario para hacer que un disquete pueda realizar la carga inicial. Se necesita usar la orden COPY para copiar COMMAND.COM.

Por ejemplo:

```
A>SYS B:

A>COPY COMMAND.COM B:
```

Esta combinación de órdenes copia las pistas de carga inicial, los ficheros del sistema y COMMAND.COM al nuevo disco.

Sintaxis

La sintaxis de la orden SYS es

SYS *d:*

El argumento *d:* es el disco que tiene que recibir las pistas de carga inicial y los ficheros del sistema.

Orden CHKDSK
Externa

La orden CHKDSK informa del tamaño del disco, de la cantidad de espacio libre en el disco y de la cantidad de RAM disponible. También avisa y a veces repara muchos errores internos del disco. Sin embargo, usted debería usar la orden RECOVER en lugar de CHKDSK para resolver los mensajes de error de «Bad sector» («Sector en mal estado»).

Usos más comunes

Ocasionalmente el MS-DOS revuelve el mapa del directorio interno que indica qué ficheros están en el disco y dónde se encuentran. Esto ocurre a veces mientras el MS-DOS está escribiendo información en un disco. Poco se puede hacer para evitarlo. Sin embargo, si ocurre, puede usar la orden CHKDSK para averiguar cuantos desperfectos se han producido y, posiblemente, arreglarlos. La orden se da usualmente sin argumentos.

Debería usar la orden CHKDSK regularmente para verificar la corrección interna de los discos importantes. Si tiene un disco duro, debería comprobar el disco con la orden CHKDSK cada vez que hace la carga inicial del sistema. Para hacer esto, ponga una orden CHKDSK en su fichero AUTOEXEC.-BAT. La mayoría de las veces, la orden CHKDSK se da sin argumentos.

Por ejemplo:

```
A>CHKDSK
Volume APPOINTMENT created Nov 20, 1983 1:49p

   179712 bytes total disk space
    22016 bytes in 3 hidden files
```

```
  53760 bytes in 5 user files
 103936 bytes available on disk

 589824 bytes total memory
 170288 bytes free
```

El listado del ejemplo previo contiene 4 elementos importantes de información:

- Puesto que CHKDSK no informó de errores, se puede asumir que el directorio del disco no tiene errores internos. Esto no quiere decir que no haya sectores en mal estado en el disco; sólo indica que el directorio está completo y que todas las entradas son consistentes. Los sectores en mal estado sólo se detectan cuando se intenta leer un fichero que está en la parte del disco que contiene estos sectores.
- La primera línea indica el rótulo del volumen (APPOINTMENT) y cuando se creó dicho rótulo.
- El siguiente conjunto de líneas da el espacio total del disco (que depende de cómo se formatee el disco y de si es de una sola cara o de doble cara), el número y tamaño total de los ficheros ocultos (descritos anteriormente en la orden DIR), el número y el tamaño de los ficheros de usuario (que son los ficheros normales del disco) y la cantidad de espacio que queda en el disco. En este caso, informa que el disco puede contener 179.712 bytes, que hay 3 ficheros ocultos y 5 ficheros normales y que hay 103.936 bytes libres en el disco.
- El último conjunto de líneas informa de la RAM total de la computadora y de la cantidad que está disponible actualmente. La diferencia entre estos dos números representa la cantidad de RAM que usa el MS-DOS junto con la que usan otros *programas residentes.* (Los programas residentes son aquellos que se cargan a sí mismos en memoria y permanecen en ella, como por ejemplo, la orden PRINT y los programas de disco-RAM). En este caso, hay 170.288 bytes de RAM libre de un total de 589.824 que hay en la computadora.

Si está usando subdirectorios, el programa CHKDSK también le informará sobre el espacio del disco utilizado por las entradas del directorio.

Por ejemplo,

```
A>CHKDSK

 179712 bytes total disk space
  22016 bytes in 2 hidden files
   1024 bytes in 2 directories
```

```
   91648 bytes in 8 user files
   65024 bytes available on disk

  589824 bytes total memory
  171840 bytes free
```

Observe la línea de más, que indica que dos directores han ocupado 1.024 bytes.

Uso de CHKDSK para probar si un fichero está fragmentado. Se puede usar la orden CHKDSK para saber si un fichero es contiguo o no. Un *fichero contiguo* es aquel que tiene todas sus partes en sectores adyacentes. La sintaxis para esta forma de la orden es

CHKDSK *especifichero*

En este caso, el nombre del fichero puede tener caracteres comodín.

La figura 3-2 muestra una pista del disquete colocada sobre una línea recta, indicando los contenidos de cada sector de la pista. El fichero SHOW.MEM ocupa los sectores 1, 2 y 3 y BAR.TXT ocupa los sectores 4 y 5; los últimos tres sectores de esta pista están vacíos. Suponga que usted usa su procesador de texto y añade texto a SHOW.MEM de forma que el fichero tiene ahora 4 sectores. ¿Dónde se coloca el nuevo sector?

El MS-DOS no pondrá los nuevos datos en el sector 4, puesto que esto borraría parte de BAR.TXT. En vez de esto, el MS-DOS permite que el fichero esté *fragmentado* o dividido en partes. En este caso, el MS-DOS vería que el sector 6 está vacío y pondría en él el cuarto sector de SHOW.MEM, resultando un mapa de sectores similar al que aparece en la figura 3-3. Usted no necesita preocuparse de cómo el MS-DOS sabe que las diferentes partes del fichero están en diferentes partes del disco; el MS-DOS maneja esto de una forma completamente natural.

Si muchos de los ficheros de un disco están fragmentados, el acceso será más lento, puesto que las cabezas del disco tienen que moverse hacia atrás y hacia adelante, más a menudo. Si tiene un disquete con muchos ficheros fragmentados, el mejor remedio es formatear un disquete nuevo y usar la orden

1	2	3	4	5	6	7	8
SHOW.MEM 1	SHOW.MEM 2	SHOW.MEM 3	BAR.TXT 1	BAR.TXT 2	(Vacío)	(Vacío)	(Vacío)

Figura 3-2. Pista de un disco que muestra ficheros contiguos

1	2	3	4	5	6	7	8
SHOW.MEM 1	SHOW.MEM 2	SHOW.MEM 3	BAR.TXT 1	BAR.TXT 2	SHOW.MEM 4	(Vacío)	(Vacío)

Figura 3-3. La adición de un nuevo sector produce un fichero fragmentado

COPY para copiar todos los ficheros en el nuevo disquete. No use la orden DISKCOPY para eliminar los ficheros fragmentados, porque simplemente copia cada sector.

Por ejemplo, para ver si el fichero CORPINV.DAT está fragmentado, escriba la orden

```
A>CHKDSK CORPINV.DAT
Volume APPOINTMENT created Nov 20, 1983 1:49p

   179712 bytes total disk space
    22016 bytes in 3 hidden files
    53760 bytes in 5 user files
   103936 bytes available on disk

   589824 bytes total memory
   170288 bytes free

All specified file(s) are contiguous.
```

Esto significa que el fichero es contiguo. Si el fichero no hubiera sido contiguo, el mensaje sería

```
CORPINV.DAT
  Contains 2 non-contiguous blocks.
```

Uso de CHKDSK con argumentos. La orden CHKDSK puede encontrar un gran número de diferentes tipos de errores. La versión 2 del MS-DOS permite usar el argumento /F para corregir algunos de esos errores. Este procedimiento puede hacer que se pierda alguna información del disco; sin embargo, si no se corrige el disco se puede perder aún más información. Por esta razón, si la orden CHKDSK le informa de que existen errores, debería ejecutarla de nuevo con el argumento /F. Observe, sin embargo, que la versión 1

del MS-DOS corrige automáticamente el disco y no hay argumento /F. La sintaxis para esta forma de la orden es

CHKDSK /F

A veces, la orden CHKDSK encuentra en el disco unidades de información que han perdido su asignación (estas unidades también se llaman grupos perdidos). Se pueden recuperar estos grupos perdidos usando la orden CHKDSK /F. La orden CHKDSK pondrá estos grupos en nuevos ficheros donde usted pueda examinarlos. Debería usar un editor de texto para ver si alguna información contenida en el grupo es texto. Si encuentra texto puede trasladarlo a los ficheros correctos. Si el grupo contiene programas, usted no podrá leerlos con un editor de texto. La información recuperada se guarda en ficheros llamados FILE*nnnn*.CHK, donde *nnnn* es un número comenzando por 0000.

El argumento /V (verboso) se usa a menudo para examinar de nuevo el disco, después de haber encontrado errores con el CHKDSK. Este argumento hace que el MS-DOS informe de los nombres de los ficheros que está examinando en el directorio, mientras avanza. Si hay errores en el disco, este argumento permite ver dónde los ha encontrado la orden CHKDSK.

Si tiene un fichero AUTOEXEC.BAT, es una buena idea incluir en él la orden CHKDSK. De esta forma, la computadora comprueba la integridad del disco cada vez que se hace la carga inicial del sistema.

Advertencias y errores frecuentes

El argumento /F de la orden CHKDSK puede ocasionar una pérdida de información que es legible por el MS-DOS. A causa de esto, siempre se debería ejecutar primero la orden CHKDSK sin el argumento /F. Cuando CHKDSK informa de errores en el disco, debería intentar inmediatamente copiar todos los ficheros fuera del disco, usando la orden COPY. Quizá pueda hacerlo o quizá no, dependiendo de la importancia de los errores. Si llega a ficheros que no puede copiar, simplemente sálteselos. Después de haber copiado todos los ficheros que haya podido, utilice la orden CHKDSK con el argumento /F.

Sintaxis

Con la orden CHKDSK se pueden usar los siguientes tipos de sintaxis:

CHKDSK [*d*:] [/F] [/V]

o

CHKDSK *especifichero*

El argumento *d:* es la unidad de disco que se quiere examinar. El argumento / F indica que se quiere que CHKDSK arregle el disco si encuentra errores. En la versión 2 del MS-DOS, normalmente CHKDSK sólo informa de los errores.

El argumento / V indica que usted quiere que la orden CHKDSK trabaje en su modo verboso y le dé mensajes completos mientras analiza el disco.

Si se da un *especifichero*, la orden CHKDSK comprueba si el fichero o ficheros están fragmentados.

Orden RECOVER
Externa
Versión 2

La orden RECOVER se usa para restaurar ficheros que están en un disco que tiene sectores en mal estado o, lo que es lo mismo, sectores dañados. Esta orden recupera los ficheros que tienen parte de sus datos en los sectores dañados, pero no puede usar RECOVER para restaurar un fichero que se ha borrado, ni se puede usar para recuperar un programa que tiene un sector dañado. (Un programa no funciona si le falta un trozo). La orden RECOVER también recupera discos que tienen parte de sus directorios en sectores dañados.

Usos más comunes

La sintaxis general es

RECOVER *especifichero*

Si le sale un mensaje de «Bad sector» («sector en mal estado») mientras está ejecutando un programa de aplicación, probablemente significa que el material del disco (la capa magnética que recubre el disco) se ha dañado. El polvo y las huellas dactilares son las causas más frecuentes de esto, en los disquetes flexibles.

El capítulo 5 trata de cómo cuidar los discos y de cómo recuperarlos cuando hay errores que ni CHKDSK ni RECOVER pueden solucionar. El capítulo 5 tiene también información sobre los programas comerciales que realizan las mismas funciones que la orden RECOVER. Algunos de estos programas tienen formas creativas de recuperar información que tienen menos probabilidad de perder información.

Generalmente, cuando sale un mensaje de error «Bad sector», se sabe a qué fichero se refiere. Por ejemplo, si mientras está escribiendo un fichero con el procesador de texto le sale un mensaje «Bad sector», usted puede asumir que probablemente es ese fichero el que tiene el sector erróeno. Para recupe-

rar tanta información del fichero como sea posible, dé la orden RECOVER con el nombre del fichero.

Por ejemplo, para recuperar el fichero TIME.TBL, dé la orden

```
A>RECOVER TIME.TBL
```

Recuerde que la orden RECOVER no recupera ninguna información del sector dañado. Por esto, no se sorprenda, si a un fichero recuperado le falta una buena cantidad de texto por en medio.

No hay forma de recuperar un programa que tenga un sector dañado porque los pasos importantes del programa, que estuvieran en dicho sector, se habrán perdido. Similarmente, es muy improbable que se pueda hacer algo para salvar una base de datos con un sector dañado, puesto que la mayoría de los programas de bases de datos almacenan los datos en una forma compacta que es similar a la de los ficheros de programas.

Si el directorio de su disco tiene un sector dañado, puede usar la orden RECOVER para intentar juntar las partes del directorio de nuevo; sin embargo, ninguno de los ficheros tendrá su nombre correcto. Cada fichero en el disco recuperado tendrá el nombre FILE*nnnn*.REC, donde *nnnn* es un número empezando con 0000. Para reconstruir el directorio, tiene que determinar el contenido de cada fichero individualmente, y darle de nuevo el nombre apropiado.

Por ejemplo, si el directorio del disco B: tiene un sector dañado, la orden para recuperar el directorio es

```
A>RECOVER B:
```

Puesto que no hay forma de que la orden RECOVER sepa qué partes del directorio del disco están en sectores válidos, esta forma de RECOVER reconstruirá todo el directorio (sin los nombres correctos de los ficheros). Este proceso es complicado si tiene un disco duro con muchos ficheros, porque descifrar cuál es cuál puede ser increíblemente lento. Por ello, asegúrese de que tiene un sector dañado en el directorio antes de usar esta forma de la orden RECOVER.

Advertencias y errores frecuentes

Recuerde que una vez que haya usado la orden RECOVER en un fichero o directorio, no hay forma de volverse atrás. Cualquier información perdida en el proceso de recuperación de un fichero, se pierde para siempre.

Sintaxis

La sintaxis de la orden RECOVER es

RECOVER *especifichero*

o bien

RECOVER *d:*

La primera forma de la orden se usa para recuperar ficheros con sectores dañados. Es muy poco probable que usted use la segunda forma de la orden RECOVER, que intenta recuperar discos que tienen un sector dañado en el directorio.

Orden DISKCOPY
Externa
Sólo IBM

La orden DISKCOPY copia un disquete entero en otro disquete, borrando el contenido previo del disquete de destino. Copiar discos con la orden DISKCOPY es usualmente más rápido que usando la orden COPY. Muchos fabricantes tienen órdenes similares.

Usos más comunes

La orden DISKCOPY proporciona un método rápido para copiar disquetes enteros.

La sintaxis general es

DISKCOPY *disco-origen: disco-destino:*

En vez de leer fichero por fichero, como la orden COPY, DISKCOPY lee toda la información de los sectores del disquete origen y, a continuación, escribe dichos sectores en el disquete de destino. (Si el disquete de destino no está formateado o tiene un formato diferente que el disco origen, la orden DISKCOPY lo vuelve a formatear). De este modo, hace una reproducción exacta, copiando las pistas de carga inicial, el directorio y los ficheros.

Por ejemplo, si tiene dos unidades de disco flexible, debería ejecutar el programa con la siguiente orden:

```
A>DISKCOPY A: B:
```

En este ejemplo, el disquete que usted quiere copiar (origen) está en A: y el disquete que recibe la copia (destino) está en B:. Puesto que toda la información del disquete de destino se borra, usted debería estar absolutamente seguro de que no los ha intercambiado. El programa le pedirá ahora que inserte el disquete origen en A: y el disquete destino en B:, con el siguiente mensaje:

```
Insert source diskette in drive A:

Insert target diskette in drive B:

Strike any key when ready
```

La orden DISKCOPY le está dando una última oportunidad de comprobar que los disquetes están en las unidades adecuadas. Pulse cualquier tecla cuando esté preparado y el programa escribirá

```
Copying 9 sectors per track, 2 side(s)
```

que significa que está copiando 9 sectores por pista y con el formato de 2 caras. Si está copiando de un disquete que está formateado con 8 sectores por pista o con 1 sola cara, los números de este mensaje serán, naturalmente, diferentes.

Si el disquete de destino no está formateado o lo está de forma diferente que el disquete origen, DISKCOPY lo formateará mientras copia y le dirá

```
Formatting while copying
```

Esto quiere decir que no tiene que formatear el disquete de destino antes de copiar. En realidad, formatear mientras se copia es más rápido que formatear primero y copiar a continuación. Usted oirá las cabezas de las unidades de disco flexible al moverse y, después de unos momentos, el programa le dirá

```
Copy complete

Copy another (Y/N)?
```

Si quiere hacer otra copia, pulse Y; en caso contrario, pulse N.

Si tiene sólo una unidad de disquete, también puede usar la orden DISKCOPY. El programa leerá primero la información del disquete origen y, a continuación, le dirá que introduzca el disquete de destino. Dependiendo de la cantidad de memoria que tenga su computadora, puede tener que intercambiar los disquetes dos o tres veces.

Por ejemplo, para usar las orden DISKCOPY en una sola unidad, escriba la orden

```
A>DISKCOPY

Insert source diskette in drive A:
Strike any key when ready
Copying 9 sectors per track, 2 side(s)
Insert target diskette in drive A:
Strike any key when ready
Copying 9 sectors per track, 2 side(s)
```

Cuando aparezca el mensaje «Insert source...», usted debe introducir el disquete origen en la unidad A: y pulsar cualquier tecla cuando esté dispuesto. Cuando aparezca «Insert target...», debe introducir el disquete destino en A: y pulsar cualquier tecla cuando esté listo.

Aunque DISKCOPY verifica los datos después de escribirlos, puede que usted quiera ejecutar la orden DISKCOMP después de copiar un disquete con DISKCOPY.

Puesto que DISKCOPY copia todo el disquete, es normal usar esta orden para hacer copias de seguridad de disquetes que contienen software. DISKCOPY copia todos los ficheros e incluso los ficheros ocultos que el fabricante haya puesto en el disco. Sin embargo, DISKCOPY no funciona con los disquetes protegidos contra copia (vea en el capítulo 5 una discusión de cómo hacer copias de seguridad de estos disquetes).

Advertencias y errores frecuentes

Como con las demás órdenes que pueden borrar datos del disquete, hay que estar seguro de tener el disquete correcto en la unidad correcta antes de ejecutar DISKCOPY. Otro método bueno para evitar borrar información de un disco equivocado es poner una etiqueta de protección de escritura en el disco origen antes de ejecutar DISKCOPY.

Sintaxis

La sintaxis de la orden DISKCOPY es

DISKCOPY [*d1:*] [*d2:*] [/1]

El argumento *d1:* es el disquete origen y *d2:* el disquete destino. Se puede especificar la misma unidad o diferentes unidades. Si se da sólo una unidad, se

asume que la unidad de destino es la unidad por defecto. Si no se da ninguna unidad, DISKCOPY asume que se quieren copiar los disquetes usando sólo una unidad. La orden DISKCOPY puede funcionar con una sola unidad; simplemente le pide a usted que introduzca alternativamente los dos disquetes. La tabla 3-3 da ejemplos de uso de estos argumentos.

El argumento / 1 indica que sólo se quiere copiar una cara del disquete, independientemente de si el disquete es de una o dos caras.

Orden DISKCOMP
Externa
Sólo IBM

La orden DISKCOMP compara dos disquetes, sector a sector para asegurarse de que son exactamente iguales. Sólo es útil inmediatamente después de haber dado una orden DISKCOPY. La razón de esto es que cualquier cambio que usted haga en un disquete (como modificar un fichero) se descubrirá con la orden DISKCOMP.

Usos más comunes

Probablemente, la única vez que usted use DISKCOMP será después de copiar discos con DISKCOPY. La sintaxis usual es

DISKCOMP *d1: d2:*

Tabla 3-3. Argumentos posibles para la orden DISKCOPY

Orden	Significado
DISKCOPY A: B:	Copia del disquete que está en A: al disquete que está en B:.
DISKCOPY A: A:	Copia del disquete que está en A: a otro disquete que se colocará en A:.
DISKCOPY B:	Copia del disquete que está en B: al disquete que está en la unidad por defecto.
DISKCOPY	Copia del disquete que está en la unidad por defecto a otro disquete que se colocará en la unidad por defecto.

Por ejemplo, si usted copió de A: a B:, puede comparar los disquetes para asegurarse de que la copia es exacta. Se usaría la siguiente orden:

```
A>DISKCOMP A: B:
```

Si los discos coinciden exactamente (como casi siempre ocurre), el programa imprime

```
Diskettes compare ok
```

Si los disquetes no coinciden, DISKCOMP informa de la posición (pista y cara) donde ha ocurrido el error. El informe relaciona las pistas que no coinciden exactamente:

```
Compare error(s) on
Track 04, Side 0

Compare error(s) on
Track 23, Side 1
```

Advertencias y errores frecuentes

Recuerde que la orden DISKCOMP sólo es útil después de DISKCOPY, puesto que las otras modificaciones que usted puede hacer al disquete (como, por ejemplo, cambiar un fichero o añadir otro fichero al disco) causarían muchas diferencias entre los disquetes, que aparecerían en el informe. Esto es similar a la relación entre las órdenes COPY y COMP.

Sintaxis

La sintaxis de DISKCOMP es

DISKCOMP [*d1:*] [*d2:*] [/1] [/8]

d1: y *d2:* son los disquetes de origen y destino, respectivamente. Esta orden tiene los mismos argumentos que DISKCOPY. Se puede especificar la misma unidad o unidades diferentes; si se da sólo una unidad, se asumirá que la unidad de destino es la unidad por defecto. Si no se da ninguna unidad, DISKCOMP asumirá que se quieren comparar los dos disquetes usando sólo la unidad por defecto. Los argumentos para DISKCOMP son los mismos que para DISKCOPY.

El argumento /1 indica que sólo se quiere comparar una cara del disquete, independientemente de si es un disquete de una o dos caras. El argumento /8 no se debería usar a no ser que los discos estuvieran formateados con la versión 1, que usa un formato de 8 sectores. El argumnento /8 hace que DISKCOMP compare 8 sectores por pista, aunque los disquetes contengan 9 sectores por pista.

Orden FDISK
Externa
Versión 2
Sólo IBM

El Fixed Disk (disco fijo) del IBM PC/XT se puede dividir en diferentes particiones de forma que se puedan ejecutar en él hasta cuatro sistemas operativos. Estas particiones actúan igual que si fueran discos rígidos separados: cada sistema operatino no conoce la parte del disco que pertenece a los otros sistemas operativos. La orden FDISK permite crear la partición del PC-DOS en el IBM Fixed Disk.

Usos más comunes

Además de crear la partición del PC-DOS en el disco fijo del IBM PC/XT, la orden FDISK también permite cambiar la *partición activa* (es decir, la partición que se está usando actualmente), suprimir la partición del PC-DOS, visualizar información acerca de la partición actual y usar FDISK sobre otro disco fijo. Cuando se introduce la orden FDISK, aparece un menú que permite seleccionar una de estas opciones.

La orden FDISK sólo es útil cuando usted monta por primera vez un PC/XT (y esto en el caso de que el distribuidor no le haya instalado el PC-DOS en el disco rígido). Esta orden funcionará con el Fixed Disk que trae el PC/XT, pero puede que no funcione con otros discos rígidos que se usan con el IBM PC. Además, FDISK sólo se puede usar para crear la partición del PC-DOS; se tienen que usar otros sitemas operativos para crear o borrar sus propias particiones. Si necesita usar esta orden consulte en el manual del IBM PC-DOS.

Orden BACKUP
Externa
Versión 2
Sólo IBM

La orden BACKUP se usa para hacer copias de seguridad, en disquetes, de los ficheros del Fixed Disk del IBM PC/XT. Este programa es difícil de usar y si se usa incorrectamente, puede borrar información valiosa de los discos. Los ficheros de seguridad producidos usando la orden BACKUP no se pueden usar directamente; hay que copiarlos de nuevo en el Fixed Disk utilizando la orden RESTORE. El capítulo 5 habla de otros métodos mejores para hacer copias de seguridad de los datos.

Usos más comunes

La sintaxis general de la orden BACKUP es

BACKUP *especifichero d:*

El argumento *especifichero* incluye el nombre de la unidad de Fixed Disk, si no es la unidad por defecto. Para más información, vea el manual del IBM PC-DOS.

Orden RESTORE
Externa
Versión 2
Sólo IBM

La orden RESTORE se usa para restablecer los ficheros de los que se hizo una copia de seguridad en un disquete, utilizando la orden BACKUP. La orden RESTORE reconstruye el contenido del disquete en el Fixed Disk del IBM PC/XT. Al igual que la orden BACKUP, es difícil de usar. El capítulo 5 discute otros métodos mejores para hacer copia de seguridad de los datos.

Usos más comunes

La sintaxis usual de la orden es

RESTORE *d: especifichero*

El argumento *d:* es el nombre de la unidad de disco flexible. El argumento *especifichero* incluye el nombre de la unidad de Fixed Disk si no es la unidad por defecto. Para más información, vea el manual del IBM PC-DOS.

INDICADORES DEL SISTEMA

Esta sección trata de las siguientes órdenes:

DATE	*Actualiza la fecha*
TIME	*Actualiza la hora*
VERIFY	*Indica al MS-DOS que haga doble comprobación cuando escriba en los discos*
MODE	*Cambia los parámetros de comunicación*
VER	*Visualiza el número de la versión del MS-DOS*
VOL	*Visualiza el rótulo de un disco*
PROMPT	*Cambia la petición de orden del MS-DOS*
BREAK	*Hace que el MS-DOS compruebe más a menudo si hay interrupciones de usuario*
ASSIGN	*Cambia las letras asignadas a los discos*
CTTY	*Cambia la consola a una puerta serie*
SET	*Cambia cadenas de entorno*

La mayoría de las órdenes de esta sección se usan para actualizar varios *indicadores del sistema* en el MS-DOS. Los indicadores del sistema son informaciones internas, guardadas por el MS-DOS, que indican a éste cómo quiere usted que actúe. Aunque no es necesario conocer estas órdenes, son muy fáciles de aprender y de usar.

Cada vez que usted hace la carga inicial del MS-DOS, el sistema toma muchos indicadores por defecto. Si no le gustan los valores asumidos por defecto puede usar las órdenes de esta sección para cambiarlos. Para cambiar automáticamente los indicadores, puede incluir las órdenes apropiadas en el fichero AUTOEXEC.BAT. Por ejemplo, usted podría cambiar el indicador de la petición de orden del MS-DOS (que usualmente es A >) a otra cadena de caracteres.

Las órdenes ASSIGN, CTTY y SET no se usan casi nunca.

Orden DATE
Interna

La orden DATE pone la fecha que el MS-DOS utiliza cuando actualiza ficheros. Esta orden actúa igual que la petición de fecha que usted ve cuando hace la carga inicial del MS-DOS.

Usos más comunes

Se puede dar la fecha en la línea de la orden o bien hacer que la orden DATE pida que se introduzca la fecha.

La sintaxis usual es

DATE [*nueva-fecha*]

Observe que esta orden actualiza la fecha interna que el MS-DOS guarda en RAM, no la fecha de un fichero (la fecha de un fichero se pone cuando usted escribe el fichero en el disco).

Por ejemplo, para dar una nueva fecha en la línea de la orden, introduzca

```
A>DATE 7/10/84
```

Para tener la petición de fecha de la orden DATE, introduzca

```
A>DATE
Current date is Tue 7-10-84
Enter new date:
```

También puede usted dejar sin dar la fecha pulsando RETURN, con lo cual se mantendrá la fecha que hubiera antes de dar la orden.

Algunas computadoras tienen relojes internos. En vez de actualizar la fecha y la hora manualmente, el fabricante puede haber incluido un programa, que usted puede ejecutar para leer el reloj interno y actualizar la fecha y la hora.

Reglas

La fecha tiene que introducirse como tres números separados por guiones (-) o barras (/). Las reglas completas para formatear la fecha se dieron en «Puesta en marcha del MS-DOS» en el capítulo 2.

Advertencias y errores frecuentes

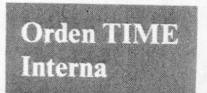
Algunos programas en BASIC cambian la fecha. Si la fecha se ha modificado incorrectamente después de ejecutar un programa, puede que usted quiera actualizarla con la orden DATE.

Sintaxis

La sintaxis para la orden DATE es

DATE [*mm-dd-aa*]

mm-dd-aa es el mes, día y año. Se puede dar esta orden con guiones (-) o barras (/) entre los números. Si no da una fecha en la línea de la orden, DATE le pedirá que la introduzca.



La orden TIME pone la hora que el MS-DOS usa cuando actualiza ficheros. Actúa como la petición de hora en la carga inicial del MS-DOS.

Usos más comunes

La sintaxis usual de la orden TIME es

TIME [*nueva-hora*]

Usted puede dar la hora en la línea de la orden o hacer que la orden TIME le interrogue.

Por ejemplo, para dar una nueva hora en la línea de la orden, introduzca

```
A>TIME 9:41
```

Para que TIME le interrogue, introduzca

```
A>TIME
Current time is 9:41:03
Enter new time:
```

Puede dejar sin dar la hora pulsando RETURN, con lo cual se deja la misma hora que hubiera antes de dar la orden.

Algunas computadoras tienen relojes internos. En vez de actualizar la fecha y la hora manualmente, el fabricante puede haber incluido un programa, que usted puede ejecutar para leer el reloj interno y actualizar la fecha y la hora.

Reglas

El único formato aceptable para la hora es en horas, minutos y segundos, separados por dos puntos (:). Las reglas completas para formatear la hora se dieron en «Puesta en marcha del MS-DOS», en el capítulo 2.

Advertencias y errores frecuentes

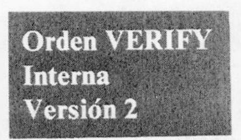
Algunos programas en BASIC cambian el reloj para cronometrar procesos. Si hacen esto, usualmente ponen la hora de medianoche (00:00:00). Así, se ve que la hora es incorrecta después de ejecutar un programa, quizá quiera actualizarla con la orden TIME.

Sintaxis

La sintaxis para la orden TIME es

TIME [*hh:mm:ss.xx*]

hh:mm:ss.xx son horas, minutos, segundos y centésimas de segundo. Hay que dar la hora usando los dos puntos (y el punto decimal si quiere mucha precisión). Los segundos y las centésimas de segundo son opcionales.



La orden VERIFY hace que el MS-DOS verifique la información cada vez que escribe en un disco. El MS-DOS comprueba la información releyendo del disco y comparando con la información que tenía que escribir.

Usos más comunes

Aunque es improbable que el MS-DOS escriba incorrectamente información en un disco, las consecuencias pueden ser desastrosas si ocurre. Las ventajas de obligar al MS-DOS a verificar lo que escribe valen más que la fracción de segundo que tarda en hacerlo. A no ser que sea crítico para usted el aumentar la velocidad de escritura en el disco, usted debería tener el indicador VERIFY en ON (indicador activado).

Para hacer que el MS-DOS verifique cuando escribe en un disco, dé la orden

```
A>VERIFY ON
```

Para averiguar si el indicador de verificación está en ON o en OFF, dé la orden VERIFY sin argumentos.

Por ejemplo, introduzca

```
A>VERIFY
VERIFY is on
```

El MS-DOS visualiza un mensaje que permite conocer el estado actual de VERIFY.

Sintaxis

La sintaxis de la orden VERIFY es

VERIFY [{ON|OFF}]

Cuando VERIFY se pone en ON, el MS-DOS verifica todos los datos que escribe. El agumento OFF le dice al MS-DOS que no verifique los datos. Como alternativa le puede decir a la orden COPY que verifique operaciones individuales de copia, utilizando el argumento /V, independientemente del estado de la orden VERIFY.

El MS-DOS, si no se le indica lo contrario, no verifica después de escribir. Si se da la orden VERIFY sin argumento, imprime su estado actual.

Orden MODE
Externa
Sólo IBM

La orden MODE cambia la forma en que el MS-DOS comunica con los dispositivos externos. Permite fijar las comunicaciones con la impresora, los adaptadores gráficos de color y monocromo y las puertas de comunicaciones.

Usos más comunes

La orden MODE tiene muchos propósitos diferentes. En general, se usa para cambiar la forma en que el MS-DOS comunica con el hardware de la computadora o con los *periféricos* (dispositivos que se añaden a la computadora). A menudo, esto es necesario si usted utiliza un hardware que usa *protocolos* de comunicaciones distintos de los que el MS-DOS asume por defecto. Los protocolos son los métodos electrónicos que el MS-DOS utiliza para hablar con los dispositivos. Los protocolos son las reglas que los dispositivos y las computadoras utilizan para comunicarse, de la misma forma que los protocolos diplomáticos en la vida real.

Cambio de los espaciamientos en la impresión. Uno de los usos de la orden MODE es cambiar los indicadores por defecto del MS-DOS, relativos a la impresora. La sintaxis usual para esta forma de la orden MODE es

MODE LPT*n: ancholínea, altolínea*

En esta forma, *n* especifica qué número de puerta de impresora (1, 2 ó 3) está usted modificando; *ancholínea* es el número de caracteres por línea que la impresora puede manejar y *altolínea* es el número de líneas por pulgada. El argumento *ancholínea* tiene que ser 80 ó 132 y *altolínea* tiene que ser 6 u 8. (Por defecto, se toman 80 y 6.)

Por ejemplo, para cambiar los indicadores por defecto de LPT1: a 132 caracteres por línea y 8 líneas por pulgada. dé la orden

```
A>MODE LPT1: 132,8
```

Hasta que usted apague la computadora o use MODE para cambiar de nuevo los indicadores, su impresora imprimirá caracteres condensados.

Si la impresora no está encendida o está funcionando, la puerta paralelo puede que no responda. En su lugar, producirá un *error de fuera de tiempo.* Usted debería utilizar el argumento P en la orden MODE, que especifica que, si la puerta no responde, el MS-DOS debe intentar enviar continuamente los datos a la puerta. Sin embargo, si la puerta no responde nunca, usted debe volver a hacer la carga inicial del sistema.

Por ejemplo, para cambiar los indicadores por defecto de LPT1: a 132 caracteres por línea y 6 líneas por pulgada, haciendo nuevos intentos continuamente, introduzca la orden

```
A>MODE LPT1: 132,6,P
```

(Observe que no se usa una barra con el argumento P.)

Ajuste de los adaptadores de imagen. La orden MODE se puede usar también para ajustar los adaptadores de imagen (las tarjetas opcionales usadas para controlar el monitor del sistema) o para pasar de un adaptador a otro. Para este uso, la sintaxis usual de la orden MODE es

MODE *anchopantalla*

La tabla 3-4 explica los posibles valores de *anchopantalla.*

Tabla 3-4. Argumentos de la orden MODE para los adaptadores de imagen

Argumento	Significado
40	Ajusta el adaptador de color a 40 caracteres de ancho.
80	Ajusta el adaptador de color a 80 caracteres de ancho.
CO40	Pasa al adaptador de color y lo pone a 40 caracteres de ancho.
CO80	Pasa al adaptador de color y lo pone a 80 caracteres de ancho.
BW40	Pasa al adaptador de color, lo pone a 40 caracteres de ancho y anula el color en la pantalla (sólo blanco y negro).
BW80	Pasa al adaptador de color, lo pone a 80 caracteres de ancho y anula el color en la pantalla (sólo blanco y negro).
MONO	Pasa al adaptador monocromo (que siempre usa 80 caracteres de ancho).

Por ejemplo, si tiene ambos adaptadores instalados y está usando actualmente el adaptador monocromo, puede pasar del adaptador monocromo al de color usando la siguiente orden:

```
A>MODE CO80
```

Si ya está usando el adaptador de color en el modo de 80 columnas y quiere pasar al modo de 40 columnas, dé la orden

```
A>MODE 40
```

No se puede cambiar el adaptador monocromo a 40 columnas.

Si está usando una tarjeta adaptadora de color, puede usar los argumentos R o L para desplazar toda la imagen en pantalla un carácter a la derecha o a la izquierda. Esto se usa para centrar la imagen en la pantalla si no está correctamente centrada (lo cual ocurre raras veces). Para comprobar si está centrada, use el argumento T (test).

Por ejemplo, si observa que no puede leer debidamente la letra de más a la derecha de la pantalla, dé la orden

```
A>MODE 80, L, T
```

El MS-DOS moverá la imagen un carácter a la izquieda y comprobará si está centrada.

Ajuste de los protocolos de comunicaciones. Otro uso de la orden MODE consiste en ajustar los protocolos de las puertas de comunicación o puertas serie. Los dispositivos que se conectan usualmente a las puertas de comunicaciones son las impresoras de tipo serie, los modems y los terminales.

Antes de conectar un dispositivo a una puerta serie, compruebe el manual para determinar qué protocolo espera el dispositivo. Si el protocolo es diferente, usualmente usted puede ajustarlo en el dispositivo mediante conmutadores. (Consulte el manual del hardware para más información de cómo hacerlo.) Los protocolos para las puertas de comunicaciones son la velocidad en baudios, la paridad, los bits de datos y los bits de parada. Estos términos se describirán en breve.

La sintaxis general para la puerta de comunicaciones es

MODE COM*n: baudios, paridad, bitsdatos, bitsparada*

La *n* indica el número de la puerta de comunicaciones. Todos los argumentos después de los *baudios* son opcionales, pero usualmente se especifican a pesar de todo. La lista que sigue define cada uno de estos argumentos. Observe que sólo pueden tomar los valores descritos en la tabla 3-5.

- *Velocidad en baudios.* Es el número de bits por segundo en la transmisión. La tabla 3-6 lista algunas velocidades corrientes para muchos dispositivos.
- *Paridad.* Es la forma de comprobar que cada byte de información se transmitió correctamente. La paridad puede ser par, impar o ninguna (si el manual del dispositivo dice que «ignora» la paridad, es lo mismo que no tener ninguna).
- *Bits de datos.* Es el número de bits, siempre 7 u 8, que contienen datos en cada byte.
- *Bits de parada.* Es el número de bits, siempre 1 ó 2, que se transmiten después de los bits de datos en cada byte.

Tabla 3-5. Argumentos de la orden MODE para la puerta de comunicaciones

Argumento	Valores posibles
baudios	110, 150, 300, 600, 1.200, 2.400, 4.800, 9.600
paridad	N (ignora paridad) O (paridad impar) E (paridad par)
bits de datos	7,8
bits de parada	1,2

Tabla 3-6. Velocidades normales de los dispositivos en baudios

Dispositivo	Velocidades normales (en baudios o bits por segundo)
Modem	300 ó 1.200 (raramente 110).
Impresora o trazador de gráficos	4.800, 9.600 19.200 (la mayoría se pueden poner a cualquier velocidad).
Terminal	9.600, 19.200 (la mayoría se pueden poner a cualquier velocidad).

Para averiguar qué valores usa su dispositivo lea el manual de instrucciones de éste. Si aún así, no puede descifrar los valores de su dispositivo, intente usar los más frecuentes: sin paridad, 8 bits de datos y 1 bit de parada. El MS-DOS asume por defecto 7 bits de datos, 1 bit de parada y paridad par.

Por ejemplo, suponga que quiere ajustar una puerta de comunicaciones (COM2:) a 9600 baudios, sin paridad, 8 bits de datos y 1 bit de parada (valores típicos para muchas impresoras de tipo serie). Usted daría la orden

```
A>MODE COM2: 9600,N,8,1
```

Cuando cambie algunos de los protocolos, use la coma para mantener las partes que quiere que permancezcan iguales. Así, si tiene una orden MODE con muchos argumentos y quiere dejar algunos de los valores actuales sin tocar, simplemente incluya las comas en los lugares de estos argumentos. Tiene que incluir una coma por cada argumento que uste se salte.

Por ejemplo, la orden

```
A>MODE COM2: 9600,,,2
```

deja la paridad y los bits de datos tal y como estaban originalmente, pero cambia los bits de parada a 2.

Como al ajustar las puertas de tipo paralelo, puede decirle al MS-DOS que ignore los errores de fuera de tiempo, añadiendo un argumento P al final de esta forma de la orden MODE.

Salida de impresora redirigida. Otro uso de la orden MODE consiste en redirigir la salida de impresora a una puerta de comunicaciones. Esto significa que cada vez que una orden envíe un fichero a la impresora (PRN:), el dato será redirigido en su lugar hacia una puerta de comunicaciones (como por ejemplo COM1:). Esto es especialmente útil si tiene una impresora serie. La sintaxis para esta forma de la orden MODE es

MODE LPT*n*:=COM*n*

Por ejemplo, si quiere que toda la información que el MS-DOS enviaría normalmente a la puerta de impresora paralelo (PRN:, que es también LPT1:) sea enviada a la puerta serie COM1:, escriba la orden

```
A>MODE LPT1:=COM1:
```

Sintaxis

La orden MODE utiliza los siguientes tipos generales de sintaxis:

Para ajustar las puertas de impresora:

MODE LPT*n*: [*ancholínea*] [,[*altolínea*] [,P]]

Para los adaptadores de imagen de color o nonocromo:

109MODE *anchopantalla* [, {R | L} [,T]]

Para las puertas serie:

MODE COM*n*: *baudios* [,*paridad* [,*bitsdatos* [,*bitsparada* [,P]]]]

Para redirigir la salida de impresora:

MODE LPT*n*:=COM*n*

La *n* es un número en todas las formas de las órdenes.

En la primera sintaxis (para las puertas de impresora), *ancholínea* es el número de caracteres, 80 ó 132, de la línea; *altolínea* es el número de líneas, 6 u 8, por pulgada. El argumento P especifica una repetición continuada del intento de comunicación. Por defecto se asume 80 caracteres por línea, 6 líneas por pulgada y sin repetir el intento.

En la segunda sintaxis (para comunicar con los adaptadores de imagen de color o monocromo), *anchopantalla* puede ser 40, 80, BW40, BW80, CO40, CO80 o MONO. Si está usando una tarjeta de adaptador de color, los argumentos R y L desplazan la imagen hacia la derecha y la izquierda, respectivamente. Puede comprobar si está correctamente alineada usando el argumento T, que pone un patrón de prueba en la pantalla.

En la tercera sintaxis (para las puertas serie), los argumentos son la velocidad en baudios, la comprobación de paridad y el número de bits de datos y bits de parada para la puerta. El argumento P es similar al argumento P utilizado en la sintaxis de la puerta de impresora porque instruye al MS-DOS para que repita el intento si la puerta no responde. Por defecto se asume E (paridad par), 7 (bits de datos) y 1 (bit de parada); si pone la velocidad en baudios a 110, por defecto el número de bits de parada será 2.

En la cuarta sintaxis (para redirigir la salida de impresora), las dos e*n*es son los números de las puertas de impresora y serie, respectivamente.

Orden VER
Interna
Versión 2

La orden VER visualiza el número de la versión del MS-DOS.

Usos más comunes

Esta orden se usa si se olvida la versión del MS-DOS que se está usando. Para usarla, simplemente introduzca VER y pulse la tecla RETURN.

Por ejemplo, en un IBM PC ejecutando la versión 2 del MS-DOS, esta orden visualizaría

```
A>VER

IBM Personal Computer DOS Version  2.00
```

Orden VOL
Interna
Versión 2

La orden VOL visualiza el rótulo de volumen de un disco. Los rótulos de volumen se ponen con la orden FORMAT, descrita anteriormente en este capítulo.

Usos más comunes

La orden VOL es extremadamente fácil de usar. La sintaxis general es

VOL [*d:*]

donde *d:* es el nombre de la unidad del disco en cuestión.

Por ejemplo, para ver el rótulo de B:, escriba la orden

```
A>VOL B:
 Volume in drive B is TRUCK1A
```

Si el volumen no tiene rótulo, la orden VOL imprimirá

```
Volume in drive B has no label
```

Sintaxis

La sintaxis para la orden VOL es

VOL [*d:*]

El argumento *d:* es la unidad que contiene al disco cuyo rótulo se quiere ver. Si no se especifica una unidad, aparecerá el rótulo de la unidad por defecto.

Orden PROMPT
Interna
Versión 2

La orden PROMPT sustituye la petición de orden del MS-DOS por cualquier otra cadena de caracteres. La petición de orden usual es la letra de la unidad por defecto, seguida por un signo de mayor que (por ejemplo, A >). Si otras personas usan su computadora, es improbable que usted quiera cambiar la petición de orden de su sistema.

Usos más comunes

Se puede usar la orden PROMPT para cambiar la petición de orden (prompt) del MS-DOS. Al cambiar la petición de orden, se puede conseguir tener más información en la pantalla cada vez que sale dicha petición de or-

Tabla 3-7. Caracteres especiales en la cadena de petición de orden

Caracter *	Significado
t	Hora actual
d	Fecha actual
n	Unidad por defecto
p	Camino actual (incluyendo la unidad)
v	Número de la versión del MS-DOS
g	Carácter mayor que (>)
l	Carácter menor que (<)
b	Carácter barra vertical (|)
q	Signo igual (=)
_	Caracteres de vuelta de carro y cambio de línea
h	Retroceso, que borra el último carácter
e	Carácter ESCAPE (código ASCII 27)
$	Carácter $

* Los caracteres especiales tienen que ir precedidos por un $ cuando se introducen en la línea de la orden PROMPT.

den. La tabla 3-7 lista los caracteres especiales que usted puede usar para añadir información a la petición de orden. Para utilizar un carácter especial, en la cadena argumento, tiene que ir precedido por un signo de dólar ($). También se puede incluir cualquier otro texto en la petición de orden. La sintaxis general es

PROMPT *cadena-petición-orden*

Suponga que quiere averiguar la fecha y la hora cada vez que aparece la petición de orden del MS-DOS. Para que la petición de orden fuera algo parecido a esto

```
17:02:33.91 ¦ Tue 7-10-84 ¦ A>
```

(es decir, añadiendo la fecha y la hora), habría que dar la orden

```
A>PROMPT $t $b $d $b $n$g
17:02:33.91 ¦ Tue 7-10-84 ¦ A>
```

Observe que hay espacios entre todos los caracteres y signos salvo entre $n y $g. Un espacio en la cadena argumento causa un espacio en la petición de orden, como se puede ver en la línea que está debajo de la orden, que muestra la nueva y más detallada petición de orden. Observe que cualquier carácter no precedido por un signo de dólar se interpreta como un carácter normal sin ninguna función especial, como por ejemplo «Hi, Sally!» («¡Hola, Sally!»).

Por ejemplo, si quiere añadir una petición de orden amistosa, que combine un mensaje con caracteres especiales, introduzca

```
A>PROMPT Hi, Sally!$_You are in $p$_Your command $q
Hi, Sally!
You are in a:\reports\sally
Your command =
```

Observe que los caracteres que hay después del primer y tercer signo de dólar son subrayados.

Reglas

Cada carácter especial en la cadena de petición de orden tiene que ir precedido por un signo de dólar. Si la orden PROMPT no reconoce el carácter, lo ignorará aunque vaya precedido por un signo de dólar.

Advertencias y errores frecuentes

Si su computadora la utiliza mucha gente, asegúrese de que los principiantes sepan qué hacer cuando vean una petición de orden distinta de A >. Una petición de orden modificada en un fichero AUTOEXEC.BAT puede ser desconcertante si otra persona pone en marcha la computadora. Sin embargo, los cambios sencillos en la petición de orden (como por ejemplo, añadir la fecha y la hora en otras líneas) no causarían, probablemente, mucha confusión. Pruebe siempre cualquier cambio en la petición de orden antes de introducirlo en un fichero AUTOEXEC.BAT.

La orden PROMPT escribe información en el *entorno,* que es una parte especial de la RAM que el MS-DOS utiliza para guardar información (vea la orden SET para más información sobre este concepto). Sin embargo, el entorno tiene un espacio limitado. Si da usted la orden PROMPT y le sale el mensaje de ERROR «Out of environment space» («Fuera del espacio del entorno»), no puede cambiar la petición de orden hasta que vuelva a hacer la carga inicial de la computadora.

Sintaxis

La sintaxis de la orden PROMPT es

PROMPT [*cadena-petición-orden*]

La *cadena-petición-orden* es una cadena de caracteres que contiene la información que se quiere que aparezca en la petición de orden del MS-DOS. Si no se da una *cadena-petición-orden* la petición de orden volverá a tomar su valor por defecto.

Orden BREAK
Interna
Versión 2

La orden BREAK hace que el MS-DOS compruebe si se ha pulsado CONTROL-C o la tecla BREAK. (Esta tecla varía de una computadora a otra; en el IBM PC es la combinación CONTROL-BREAK). Esta orden es útil si se están ejecutando programas que sólo hacen cálculos y gestión de memoria, sin imprimir caracteres en la pantalla ni leer el teclado.

Usos más comunes

BREAK puede ponerse en ON o en OFF (por defecto está en OFF). El argumento ON le dice al MS-DOS que, siempre que un programa ejecute alguna orden interna, debería comprobar si la combinación de teclas CONTROL-BREAK ha sido pulsada. El argumento OFF le dice al MS-DOS que lo compruebe sólo cuando un programa realiza un acceso a la pantalla o al teclado. Así, poniendo BREAK en ON se consigue que el sistema compruebe más a menudo si se ha pulsado la citada combinación de teclas que permite abortar la ejecución de un programa.

El único uso de esta orden es hacer que el MS-DOS compruebe con mayor frecuencia de lo normal si usted ha pulsado CONTROL-C o la tecla BREAK.

Por ejemplo, introduzca

```
A>BREAK ON
```

para poner BREAK en ON

Para obtener el estado actual de BREAK, dé la orden sin argumentos.

Por ejemplo, introduzca la siguiente orden para ver si BREAK está en ON o en OFF:

```
A>BREAK
BREAK is on
```

Sintaxis

La sintaxis de la orden BREAK es

BREAK [{ON|OFF}]

Las únicas posibilidades para el argumento son ON y OFF. Si la orden BREAK se da sin argumentos, imprime su estado actual.

Orden ASSIGN
Externa
Versión 2

La orden ASSIGN hace que temporalmente el MS-DOS reasigne las letras de los discos, de forma que cuando un programa pida información de un disco, realmente la obtenga de otro. Esta orden se usa raras veces y, a no ser que se utilice con cuidado, puede causar resultados desastrosos.

Usos más comunes

Es improbable que necesite utilizar esta orden, puesto que la mayoría de los programas escritos para el MS-DOS no contienen el error de programación que la orden ASSIGN puede solucionar. La orden ASSIGN sólo es útil antes de ejecutar un programa que incluye los nombres de las unidades en los ficheros de los que lee o en los que escribe. Muy pocos programas especifican el nombre de la unidad en los nombres de ficheros; usualmente, asumen que todos los ficheros están en la unidad por defecto. La sintaxis general es

ASSIGN *letra-antigua=nueva-letra*

donde *letra-antigua* representa el nombre de la unidad escrita en el programa y *nueva-letra* es el nombre de la unidad que se debería usar en su lugar.

Por ejemplo, considere lo que ocurriría si su computadora tuviera sólo las unidades A: y B:, y usted tuviera un programa de aplicación que intentara leer el fichero C:TENURE.DAT. Puesto que usted no tiene unidad C:, el programa no encontraría nunca el fichero. Sin embargo, si usted tuviera una copia del fichero en B:, podría dar la siguiente orden antes de ejecutar el programa:

```
A>ASSIGN C=B
```

Esto le dice al MS-DOS que siempre que un programa intente leer o escribir un fichero de C:, el MS-DOS debería mirar en B: en vez de en C:.

Si es necesario, se puede hacer más de una reasignación en una orden ASSIGN.

Por ejemplo, para asignar las peticiones de datos de A: y B: a la unidad C:, dé la orden

```
A>ASSIGN A=C B=C
```

El MS-DOS mirará ahora en C: para todas las peticiones de datos de A: y B:

Una vez que haya terminado el programa, asegúrese de eliminar la asignación dando la orden ASSIGN sin argumentos

Escriba la orden

```
A>ASSIGN
```

A las unidades de disco se les vuelven a asignar sus nombres habituales.

Reglas

No se pueden usar los dos puntos en la orden ASSIGN. Algunos programas, como por ejemplo las órdenes de IBM DISKCOPY y DISKCOMP ignoran las reasignaciones hechas con la orden ASSIGN

Advertencias y errores frecuentes

La orden ASSIGN puede ser peligrosa si no se usa con cuidado. Por ejemplo, usted puede hacer una asignación no deseada y los programas que usted ejecute posteriormente no buscarán los ficheros en las unidades correctas. Por ejemplo, si primero da una orden ASSIGN y, a continuación, usa la orden ERASE, puede borrar ficheros de un disco equivocado.

Otro problema con la orden ASSIGN ocurre cuando se intenta reasignar la letra del disco que contiene al programa ASSIGN. Cuando esto ocurre, no se puede eliminar la asignación hasta que se vuelve a hacer la carga inicial de la computadora, puesto que ya no se puede acceder al disco que tiene el programa ASSIGN

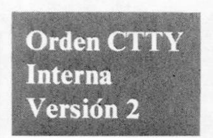
Sintaxis

La sintaxis de la orden ASSIGN es

ASSIGN [*d1=d2* ...]

d1 y *d2* son las letras de la unidad que se quiere reasignar. Si un programa pide leer o escribir en el disco *d1* después de recibir la orden ASSIGN, el MS-DOS lee o escribe realmente en el disco *d2*. No se pueden usar los dos puntos en la orden ASSIGN.

Si no se dan argumentos en la orden ASSIGN, todas las asignaciones a las unidades volverán a la normalidad.



La orden CTTY permite cambiar la consola estándar (teclado y pantalla) a una de las puertas de comunicaciones. Esta orden se usa muy raramente.

Usos más comunes

La única vez que podría usted usar esta orden sería si quisiera conectar un terminal a una de las puertas de comunicaciones. La sintaxis general de la orden es

CTTY *dispositivo*

El *dispositivo* es donde se quiere que esté la consola. Observe que no se usan los dos puntos en el nombre del dispositivo (en muchas órdenes sí se usan).

Por ejemplo, si usted conecta un terminal a la primera puerta auxiliar (que se llama AUX: o COM1:), puede transformarlo en la consola del MS-DOS dando la orden

```
A>CTTY AUX
```

o bien

```
A>CTTY COM1
```

Para volver a la consola normal, dé la orden

```
A>CTTY CON
```

Reglas

No incluya los dos puntos en el nombre del dispositivo. Los únicos dispositivos aceptables son AUX, CON y COM*n* (donde *n* es un número).

Advertencias y errores frecuentes

Una vez que haya cambiado la consola a una de las puertas de comunicaciones, usted no puede volver a pasar al estado inicial sin dar algunas órdenes a un dispositivo (como por ejemplo, un terminal) que está conectado a esa puerta. Para dar órdenes, tiene que usar la consola recién asignada. Así que, asegúrese de que el terminal que ha colgado a esa puerta tiene la velocidad en baudios y los protocolos de datos adecuados (vea el manual del dispositivo para estas especificaciones). Si usted tiene un error aquí, tendrá que volver ha hacer la carga inicial de la computadora.

Sintaxis

La sintaxis de la orden CTTY es

CTTY *dispositivo*

dispositivo es el dispositivo que se quiere que el MS-DOS tome como consola. No incluya los dos puntos en el nombre del dispositivo.



La orden SET define las *variables de entorno,* que se guardan en memoria con el MS-DOS. Algunas de estas variables se definen con otras órdenes (tales como PROMPT y PATH).

La orden SET se usa raras veces, salvo en los casos en que un programa de software específico exige que usted use dicha orden antes de ejecutar el soft-

ware. En este caso, la documentación del programa probablemente especificará la exacta orden SET que usted necesita.

Usos más comunes

No hay ningún uso corriente de la orden SET. Sin embargo, si por casualidad usted tiene un programa de aplicación que le dice que use la orden SET, es muy fácil de utilizar.

Por ejemplo, para ver todas las variables que están actualmente en memoria, utilice la orden

```
A>SET
COMSPEC=A:\COMMAND.COM
PATH=
```

Esto indica que las variables COMSPEC y PATH están definidas. El MS-DOS las define cuando se hace la carga inicial del sistema; sin embargo, el MS-DOS no hace nada con estas definiciones. Si usted quisiera añadir la variable DATDISK y darle el valor A:, daría la orden

```
A>SET DATDISK=A:

A>SET
COMSPEC=A:\COMMAND.COM
PATH=
DATDISK=A:
```

También se pueden usar las variables de entorno dentro de ficheros batch. Su uso es similar al de los argumentos de la línea de la orden. Para utilizar una variable del entorno, ponga el nombre de la variable entre signos de porcentaje, como se vio en «Utilización de argumentos dentro de los ficheros batch» en el capítulo 2.

Por ejemplo, para utilizar la variable DATDISK desde dentro de un fichero batch, usted se referiría a ella como %DATDISK%. Para escribir el fichero ACCREC.FIL del disco cuyo nombre está contenido en DATDISK, ponga la siguiente línea en el fichero batch:

```
TYPE %DATDISK%ACCREC.FIL
```

Reglas

Si usted ha cargado programas u órdenes que permanecen en memoria (como por ejemplo, PRINT), la cantidad de RAM que el MS-DOS utiliza para el entorno estará limitada a 127 bytes. Esto significa que la longitud total de todas las cadenas que usted ponga en el entorno con la orden SET está limitada. Sin embargo, puesto que muy pocos programas utilizan el entorno, probablemente usted no tendrá que preocuparse de si se sale o no del espacio permitido, cuando añade variables.

Sintaxis

La sintaxis de la orden SET es

SET [*nombrevar*=[*cadena*]]

El argumento *nombrevar* es el nombre de la variable que se quiere definir y *cadena* es el valor que se quiere dar a la variable. La orden SET convierte todas las letras de *nombrevar* en mayúsculas, pero deja *cadena* sin tocar.

Si se da un *nombrevar* sin una *cadena,* esa variable se quita del entorno MS-DOS. Esto es lo mismo que borrar la variable. Si no se dan argumentos en la orden SET, ésta listará todas las variables del entorno que estén definidas.

ORDENES DE MANTENIMIENTO DE CAMINOS

Esta sección trata las siguientes órdenes:

MKDIR (también MD)	*Añade subdirectorios a un disco*
CHDIR (también CD)	*Cambia el subdirectorio por defecto*
RMDIR (también RD)	*Suprime subdirectorios de un disco*
TREE	*Imprime la lista de subdirectorios*
PATH	*Cambia la lista de búsqueda de órdenes*

Las órdenes de esta sección permiten utilizar los directorios estructurados en forma de árbol, disponibles en la versión 2 del MS-DOS. Todas estas órdenes (llamadas frecuentemente órdenes de camino u órdenes de árbol) implican la división del directorio de un disco en muchos subdirectorios. Un camino es el método que usa el MS-DOS para encontrar su vía de paso a través de los subdirectorios. Los subdirectorios se organizan igual que las raíces de un árbol, y el subdirectorio de arriba se llama «raíz».

Si usted tiene un disco duro, estas órdenes son muy útiles, porque le permiten gestionar un gran número de ficheros del disco. Desafortunadamente, estas órdenes sólo son aplicables en la versión 2 del MS-DOS o versiones posteriores; no existen de ninguna forma en la versión 1 del MS-DOS.

Si no está familiarizado con los directorios en forma de árbol o directorios arborescentes, éstos están descritos en detalle en el capítulo 4. Probablemente, debería leer el capítulo 4 antes de leer las órdenes de esta sección, puesto que necesita entender lo que son los caminos para poder usarlos.

Orden MKDIR
Orden MD
Interna
Versión 2

La orden MKDIR se usa para crear un nuevo subdirectorio. La orden MD es idéntica a la orden MKDIR.

Usos más comunes

La orden MKDIR se puede usar para crear un nuevo subdirectorio cuando se quiere dividir más el árbol de subdirectorios. La sintaxis general es

MKDIR *camino*

Por ejemplo, si usted quisiera añadir el subdirectorio TELE a su subdirectorio actual, daría la orden

```
A>MKDIR TELE
```

o bien

```
A>MD TELE
```

MKDIR no le limita a añadir un subdirectorio sólo al subdirectorio actual. Puede usarla también para crear un subdirectorio dentro de un subdirectorio diferente.

Por ejemplo, suponga que quiere añadir el subdirectorio BOOKS al subdirectorio \MATERIAL\OFFICE. Independientemente del subdirectorio en el que esté actualmente, daría la orden

```
A>MKDIR \MATERIAL\OFFICE\BOOKS
```

La barra hacia atrás (\ ) siguiendo directamente a la orden MKDIR le dice al MS-DOS que comience su camino en el directorio principal; y las siguientes barras hacia atrás indican un camino desde un subdirectorio a otro interior a él.

También se pueden añadir subdirectorios a subdirectorios de otros discos.

Por ejemplo, para añadir el subdirectorio JONES al subdirectorio por defecto de la unidad B:, dé la orden

```
A>MKDIR B:JONES
```

Advertencias y errores frecuentes

El nombre entero del subdirectorio que usted crea no puede tener más de 63 caracteres incluyendo las barras. La razón es que el MS-DOS sólo puede guardar 63 caracteres internamente como nombres. A no ser que usted tenga más de 5 niveles de subdirectorios, la longitud del nombre no será un problema.

Sintaxis

La sintaxis de la orden MKDIR es

MKDIR [*d:*]*camino*

Si se especifica un *d:*, se creará un nuevo subdirectorio en ese disco; en caso contrario, se creará en la unidad por defecto. El argumento *camino* es el camino que se quiere crear.

Orden CHDIR
Orden CD
Interna
Versión 2

La orden CHDIR se utiliza para cambiar el subdirectorio actual o para visualizar su nombre. La orden CD es idéntica a CHDIR.

Usos más comunes

La orden CHDIR se usa frecuentemente para pasar a un subdirectorio diferente. La sintaxis usual es

CHDIR *nombre-dir*

Por ejemplo, si usted está en el subdirectorio \BASIC\TIMEACCT y quiere pasar al subdirectorio \WP\LETTERS\JIM, escriba la orden

```
A>CHDIR \WP\LETTERS\JIM
```

o bien

```
A>CD \WP\LETTERS\JIM
```

Se puede pasar al directorio raíz con la orden

```
A>CHDIR \
```

Se puede comprobar cuál es el subdirectorio actual (o por defecto), simplemente dando la orden CHDIR sin argumentos.

De este modo, después de la orden CHDIR dada en el ejemplo previo, el resultado sería

```
A>CHDIR
A:\wp\letters\jim
```

Observe que la orden CHDIR escribe el nombre del camino en minúsculas.

También se puede cambiar el subdirectorio por defecto, de otro disco. Para hacerlo, simplemente añada el nuevo nombre y el nombre de la unidad. La orden CHDIR no cambiará la unidad por defecto.

Por ejemplo, si quiere cambiar el subdirectorio actual de la unidad B: a PAYABLE.FIL dé la orden

```
A>CHDIR B:\PAYABLE.FIL
```

Para ver el nombre del subdirectorio actual en el otro disco, escriba la orden

```
A>CHDIR B:
B:\payable.fil
```

Observe que la unidad por defecto es todavía A:.

Muchos programas de aplicación del MS-DOS no pueden usar directorios arborescentes. Esto significa que no pueden tener acceso a ficheros de otros subdirectorios. Si está usando un programa de esta clase, tiene que estar seguro de que todos los ficheros a los que quiere tener acceso están en el subdirectorio actual.

Reglas

Si está cambiando el subdirectorio actual de un disco diferente, no puede incluir un espacio entre el nombre del disco y el subdirectorio.

Por ejemplo,

```
A>CHDIR B: \PAYABLE.FIL
Invalid directory
```

El MS-DOS da un mensaje de error (Directorio no válido) debido al espacio entre el nombre del disco y el subdirectorio.

Sintaxis

La orden CHDIR tiene la siguiente sintaxis:

CHDIR [[*d:*]*camino*]

o

CHDIR [*d:*]

La primera forma de la orden se usa para cambiar el subdirectorio actual en una unidad, y la segunda forma se utiliza para visualizar el subdirectorio actual. En ambas formas, *d:* es la unidad sobre la que trabaja la orden; si no se especifica, CHDIR asumirá la unidad por defecto; *camino* es el nombre del camino del subdirectorio al que se quiere pasar.

Orden RMDIR
Orden RD
Interna
Versión 2

La orden RMDIR suprime un subdirectorio de un disco si no contiene ficheros. La orden RD es idéntica a RMDIR.

Usos más comunes

La orden RMDIR se usa para eliminar un subdirectorio que no se necesita. Puesto que sólo elimina subdirectorios si no contienen ficheros, primero hay que eliminar éstos con la orden ERASE. La orden RMDIR no suprimirá un subdirectorio que contenga otros subdirectorios, ni siquiera aunque estos subdirectorios estén vacíos. La sintaxis general es

RMDIR *nombre-dir*

donde *nombre-dir* es el nombre del subdirectorio que se quiere eliminar.

Por ejemplo, si quisiera eliminar el subdirectorio SHARON del subdirectorio actual, daría la orden

```
A>RMDIR SHARON
```

o bien

```
A>RD SHARON
```

También se pueden eliminar subdirectorios de otro disco. Puede usar la orden DIR para comprobar que el directorio se ha eliminado.

Por ejemplo:

```
A>RMDIR B:BENCHMRK
```

suprime el subdirectorio BENCHMRK del disco situado en la unidad B:

Si se da un camino con más de un nombre, sólo se suprime el último subdirectorio nombrado.

Así, para eliminar el subdirectorio OVERDUE del subdirectorio \ACCOUNTS \PAYABLE, escriba la orden

```
A>RMDIR \ACCOUNTS\PAYABLE\OVERDUE
```

Reglas

La orden RMDIR no permite suprimir un subdirectorio que tenga algún fichero. Si intenta eliminar un subdirectorio de este tipo, le saldrá el mensaje «Invalid path, not directory, or directory not empty» (Camino no válido, no es directorio o directorio no vacío).

Sintaxis

La sintaxis de la orden RMDIR es

RMDIR [[*d:*]*camino*

El argumento *d:* es el disco que contiene el subdirectorio vacío que se quiere suprimir. Si se especifica un *d:*, el subdirectorio vacío se suprimirá de ese disco; en caso contrario, se suprimirá de la unidad por defecto. El argumento *camino* es el nombre del camino que se quiere eliminar. Sólo se suprime el último subdirectorio del camino.

Orden TREE
Externa
Versión 2
Sólo IBM

La orden TREE visualiza todos los caminos de los subdirectorios de un disco. Sin embargo, no visualiza la forma del árbol de subdirectorios. En cambio, utiliza caminos con barras hacia atrás para separar los subdirectorios que forman el camino.

Usos más comunes

La sintaxis usual de la orden TREE es

TREE [*d:*]

Recuerde que cada disco tiene un árbol diferente, dependiendo de qué subdirectorios se hayan creado, así que usted puede ejecutar la orden TREE para cada disco.

La figura 3-4 muestra un árbol simple. La salida de la orden TREE para este árbol tendrá el siguiente aspecto (la salida ha sido compactada para mayor claridad).

```
A>TREE
DIRECTORY PATH LISTING FOR VOLUME ???????????

Path: \HARRIS
Sub-directories:  CHRIS
                  SANDY

Path: \HARRIS\CHRIS
Sub-directories:  None

Path: \HARRIS\SANDY
Sub-directories:  None
```

```
Path: \POWERS
Sub-directories:  GENE

Path: \POWERS\GENE
Sub-directories:  None
```

Si usa la orden TREE en un disco que no tiene subdirectorios, la orden TREE responde «No sub-directories exist» (No existen subdirectorios). Esta es una forma rápida de saber si un disco tiene o no subdirectorios.

El argumento /F le indica a la orden TREE que liste todos los ficheros de cada subdirectorio.

Por ejemplo, si el subdirectorio \POWERS\GENE tuviera los ficheros LIGHTS.DOC y MAKEPROG.BAS la orden TREE /F mostraría

```
Path: \POWERS\GENE
Sub-directories:  None
Files:            LIGHTS  .DOC
                  MAKEPROG.BAS
```

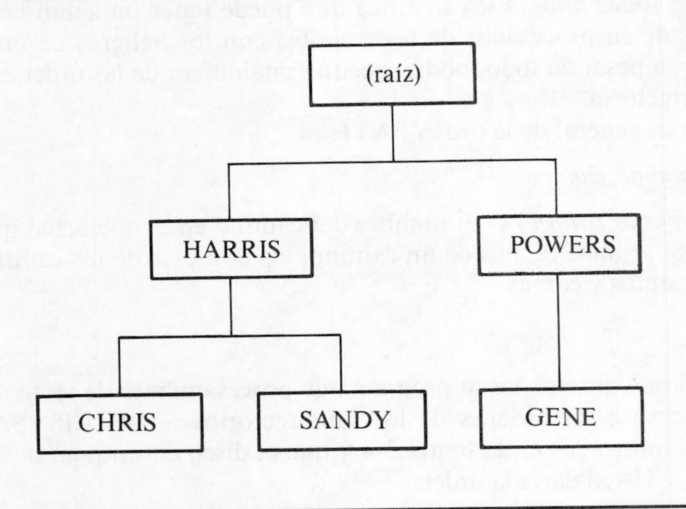


Figura 3-4. Un directorio sencillo en forma de árbol

Sintaxis

La sintaxis de la orden TREE es

TREE [*d:*] [/F]

d: es el disco que contiene los caminos que se quieren ver. Si no se especifica una unidad, se visualizarán los caminos de la unidad por defecto. El argumento / F hace que la orden TREE liste los nombres de todos los ficheros que hay en el subdirectorio.

Orden PATH
Interna
Versión 2

Normalmente, el MS-DOS busca las órdenes o los programas de aplicación en el mismo directorio donde residen los ficheros de ese programa. La orden PATH le dice al MS-DOS dónde buscar las órdenes y los ficheros batch si no puede encontrarlos en el subdirectorio actual.

Usos más comunes

Si usted guarda sus órdenes, ficheros batch o programas de aplicación en muchos subdirectores diferentes, la orden PATH le permite decir al MS-DOS dónde están todos ellos. Esto significa que puede tener un subdirectorio con las órdenes de su procesador de texto y otro con los ficheros de órdenes del MS-DOS y, a pesar de todo, poder ejecutar cualquiera de las órdenes de cualquier subdirectorio.

La sintaxis general de la orden PATH es

PATH *camino; camino ...*

El argumento *camino* es el nombre del camino en el que usted quiere que se busque. Si se incluye más de un camino, los nombres de los caminos se separan con puntos y comas.

Suponga que quiere que su programa de procesamiento de texto sea capaz de tener acceso a las órdenes de los subdirectorios \ HARRIS \ SANDY y \ POWERS mostrados en la figura 3-4 y que el disco descrito en la figura 3-4 es el disco C:. Usted daría la orden

```
A>PATH C:\HARRIS\SANDY;C:\POWERS
```

Observe que se usa un punto y coma para separar los dos caminos nombrados en la orden.

El orden en que se dan las órdenes es el mismo en que el MS-DOS buscará en los subdirectorios (con la excepción de que el MS-DOS siempre buscará primero en el subdirectorio actual).

Para ver el camino de búsqueda que está usando el MS-DOS escriba la orden PATH sin argumentos.

Por ejemplo:

```
A>PATH
PATH C:\HARRIS\SANDY;C:\POWERS
```

hace que el MS-DOS visualice el camino de búsqueda actual.

Si se quiere, se pueden mezclar unidades en una orden PATH. Sin embargo, hay que estar seguro de que todas las unidades están activas (es decir, que están preparadas para ser leídas; una unidad de disquete con su puerta abierta no está lista para que se pueda tener acceso a ella).

Por ejemplo:

```
A>PATH A:\;B:\;C:\;C:\POWERS
```

La orden PATH sólo le dice al MS-DOS dónde encontrar las órdenes y los ficheros batch, pero no otros ficheros. De este modo, no afecta a cómo el MS-DOS encuentra los ficheros de datos, los ficheros de cobertura y otros ficheros que se podrían nombrar en las órdenes. No hay forma de especificar el orden de búsqueda para estos ficheros.

Advertencias y errores frecuentes

Si el MS-DOS está buscando un subdirectorio de la lista de caminos y ese subdirectorio no existe, simplemente continuará buscando. Por eso, no le saldrá un mensaje de error si da nombres de subdirectorios incorrectos en la orden PATH.

Debería incluir siempre el nombre del disco con cada nombre de camino puesto que, sin él, el MS-DOS buscará el nombre del camino en el disco por defecto.

Por ejemplo, su orden PATH podría ser

```
B>PATH \POWERS;\
```

Si su unidad por defecto es B: y da una orden que no se encuentra en el subdirectorio actual, el MS-DOS buscará la orden en B:\POWERS y en B:\. Si usted realmente quería mirar en A:\POWERS y en A:\ debía haber dado la orden

```
B>PATH A:\POWERS;A:\
```

Sintaxis

Las sintaxis para la orden PATH son

PATH [*d:*]*camino*[;[*d:*]camino ...]

o bien

PATH

Cada parte de la lista de caminos tiene que incluir un nombre de *camino* y puede incluir opcionalmente un nombre de unidad (*d:*). Los miembros de la lista están separados por puntos y comas. No se pueden incluir espacios en la lista de caminos. Si no se dan argumentos, PATH imprime el camino actual.

Para que el orden de búsqueda vuelva a ser el asumido por defecto (sin caminos) ponga como único argumento un punto y coma.

ORDENES DE FICHEROS BATCH

Esta sección trata de las siguientes órdenes:

ECHO	*Escribe un mensaje en la pantalla*
PAUSE	*Espera a que se pulse una tecla*
IF	*Ejecuta órdenes según una decisión*
FOR	*Repite una orden para muchos argumentos diferentes*
GOTO	*Salta a una parte distinta del fichero batch*
SHIFT	*Mueve los argumentos de la línea de la orden*
REM	*Pone anotaciones en los ficheros batch*
COMMAND	*Ejecuta otro programa*

Estas órdenes se llaman *órdenes de ficheros batch* porque sólo se usan dentro de los ficheros batch.

En el capítulo 2 se vio cómo escribir ficheros batch. Resumiento el tema brevemente, los ficheros batch son ficheros que contienen una lista de órdenes que se darían normalmente al MS-DOS. Los ficheros batch pueden incluir órdenes del MS-DOS, nombres de programas de aplicación o el nombre de otro fichero batch. También se puede ver el fichero AUTOEXEC.BAT, que es un fichero batch que se ejecuta automáticamente cuando se hace la carga inicial del sistema.

Ahora que ya ha leído acerca de la mayoría de las órdenes del MS-DOS, probablemente habrá encontrado algunas otras órdenes que puede incluir en su fichero AUTOEXEC.BAT.

Un ejemplo de un fichero AUTOEXEC.BAT muy útil, que utiliza algunas de estas otras órdenes es

AUTOEXEC.BAT

```
VERIFY ON
PATH A:\DOS;A:\WP;A:\OTHERS
MODE COM1:9600,N,8,1,P
PROMPT $t$_$n$g
MODE MONO
CHKDSK
```

Las órdenes del fichero AUTOEXEC.BAT típico mostrado aquí fijarán varios de los indicadores del sistema. Este fichero pone en ON el indicador VERIFY, establece el camino de búsqueda de órdenes, ajusta la puerta de comunicaciones, cambia la petición de orden del MS-DOS para incluir la hora en unà línea aparte, elige como consola primaria el monitor monocromo y comprueba la consistencia interna del disco por defecto.

De las ocho órdenes batch presentadas en esta sección, las usadas más frecuentemente son ECHO, PAUSE, COMMAND, IF y FOR.

Orden ECHO
Interna
Versión 2

La orden ECHO permite activar o desactivar la visualización de las líneas que aparecen mientras se está ejecutando un fichero batch. También permite visualizar un mensaje en la pantalla.

Usos más comunes

La orden ECHO controla si aparece o no cada línea de un fichero batch mientras se ejecuta. Normalmente, el MS-DOS visualiza cada línea (con los argumentos substituidos) de forma que se pueda ver lo que está ocurriendo.

Si no quiere ver cada línea, incluya en el fichero batch la orden ECHO OFF. Esto se usa si hay muchas líneas en el fichero batch y no se quiere que se impriman cada vez que se ejecuta el fichero. Es en su mayor parte una cuestión de estética. Para volver a activar el indicador ECHO incluya la orden ECHO ON.

Se puede visualizar un mensaje en la pantalla utilizando la orden ECHO seguida por el mensaje.

Por ejemplo, se podría escribir un fichero batch que contuviera la orden

```
ECHO Insert the new disk in A:
```

Cuando el procesador de batch (procesador por lotes) ejecute la línea, visualizará en pantalla el mensaje «Insert the new disk en A:».

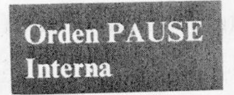
Puesto que ECHO está siempre activada cuando se están dando órdenes MS-DOS, siempre se visualizará la primera línea de cualquier fichero batch. Si no quiere que se visualice ninguna de las órdenes de un fichero batch, escriba la orden ECHO OFF en la primera línea de su fichero batch.

Sintaxis

La sintaxis de la orden ECHO es

ECHO [{ON|OFF|*mensaje*}]

El argumento ON activa la visualización de las líneas y el argumento OFF la desactiva. Si se da un *mensaje,* éste se visualiza en la pantalla. Si no se dan parámetros, la orden ECHO informa del estado del indicador ECHO.



La orden PAUSE visualiza el mensaje «Strike a key when ready...» (Pulse una tecla cuando esté preparado) y espera a que se pulse una tecla para continuar.

Usos más comunes

La orden PAUSE es útil si las operaciones que usted está a punto de realizar requieren alguna operación mecánica.

Por ejemplo, si usted crea un fichero batch para imprimir el fichero PAYROLL.RPT por la impresora, puede que quiera preguntar primeramente al usuario si la impresora está encendida. Podría añadir la frase «Be sure that the printer is turned on an ready» (Asegúrese de que la impresora está encendida y preparada) y después la orden PAUSE, como se muestra en el siguiente fichero batch.

PRN PAY.BAT

```
ECHO Be sure that the printer is turned on and ready.
PAUSE
COPY PAYROLL.RPT PRN:
```

Este fichero batch permite comprobar que la impresora está preparada antes de enviarle un fichero. Mientras se está usando la orden PAUSE, se puede hacer lo que se quiera, excepto utilizar el teclado. Para continuar el fichero batch se pulsa cualquier tecla.

Otro uso corriente de la orden PAUSE es dejar que el usuario ponga un nuevo disquete en una unidad.

Por ejemplo, un fichero que pida al usuario que inserte un disquete sería algo semejante a esto:

TOARLEEN.BAT

```
ECHO Put the diskette labelled
ECHO "Arleen's Data" in the left-hand drive
PAUSE
COPY %1 A:
```

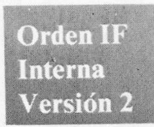
Este fichero batch le pide al usuario que ponga un disquete en una unidad antes de copiar el fichero que se nombra en el primer argumento de la orden. (Observe que se tiene que repetir la orden ECHO para un mensaje de dos líneas).

Sintaxis

La orden PAUSE se introduce como sigue:

PAUSE

Se incluye dentro de un fichero batch.



La orden IF instruye al MS-DOS para que ejecute o no ejecute órdenes según una condición variable.

Usos más comunes

La sintaxis general de la orden IF es

IF *condición orden*

La orden IF comprueba si la *condición* es verdad. Si lo es, se ejecuta la *orden;* si no lo es, se salta la *orden.* Es equivalente al uso de la palabra «si» en español: «si está lloviendo, no salgas». La tabla 3-8 lista y describe las tres formas que puede tomar el argumento *condición.*

Tabla 3-8. Condiciones utilizadas en la orden IF

Condición	Función
EXIST *especifichero*	Para comprobar si un fichero existe o no. Si se utilizan caracteres comodín en el *especifichero,* esta condición será verdad si algún fichero concuerda con la especificación.
cadena1==cadena2	Para determinar si las dos cadenas coinciden exactamente. Usualmente se utiliza para comprobar si se ha dado o no un argumento particular.
ERRORLEVEL *número*	Para comprobar si el nivel de error fijado por la última orden coincide con el número. Puesto que muy pocas órdenes actualizan el nivel de error, esta forma se utiliza raras veces.

La orden IF se puede utilizar para comprobar si un fichero existe o no, antes de ejecutar una orden. Para hacerlo, utilice la forma EXIST de la condición.

Por ejemplo, suponga que está a punto de listar el fichero NEWMSG.TXT. Utilizando la condición EXIST, puede comprobar primeramente si existe, para evitar que salga un mensaje de «File not found» (Fichero no encontrado). Incluya la siguiente orden en su fichero batch:

```
IF EXIST NEWMSG.TXT TYPE NEWMSG.TXT
```

Una aplicación frecuente de la condición EXIST consiste en comprobar si existe un fichero antes de crear otro con el mismo nombre.

Si va a cambiar de nombre MANUAL.NEW a MANUAL.OLD, debería verificar si MANUAL.OLD ya existe (si existe, debería sacar un mensaje por la pantalla). Para automatizar este proceso podría incluir la siguiente orden:

```
IF EXIST MANUAL.OLD ECHO Can't rename MANUAL.NEW
```

En este caso, si el fichero existe, la orden ECHO visualizará «Can't rename MANUAL.NEW» (No se puede cambiar dę nombre a MANUAL.NEW).

La condición basada en la igualdad de cadenas, de la orden IF (la segunda condición en la tabla 3-8) se usa para comprobar si se ha dado un argumento particular. Esto se hace comparando las dos cadenas.

Por ejemplo, usted puede necesitar que se copie el fichero TOTCOST.TXT en el disco B: si el tercer argumento dado en la línea de la orden es la palabra MOVE. Para hacer esto, usaría la orden

```
IF %3==MOVE COPY TOTCOST.TXT B:
```

Desafortunadamente, tienen que coincidir exactamente (incluyendo mayúsculas y minúsculas) y usted tiene que decirle al MS-DOS qué argumento contiene a la cadena. Usted no está acostumbrado a esta limitación en las otras órdenes del MS-DOS; usualmente, el MS-DOS no tiene en cuenta si las letras son mayúsculas o minúsculas y, a veces, no tiene en cuenta el orden de los argumentos.

También se puede usar el argumento NOT después de la orden IF para negar la condición.

De este modo, la línea

```
IF NOT EXIST MACHINE.TXT COPY B:NEWMACH.TXT
```

ejecutaría la orden COPY si el fichero MACHINE.TXT no existiera.

Un tercer uso de la orden IF es comprobar el nivel de error. En este caso, la orden IF comprueba si ciertas órdenes se han ejecutado correctamente (es el tercer caso de la tabla 3-8). Muy pocas órdenes actualizan el nivel de error, así que esta condición puede que no le sea útil.

El *nivel de error* es un número dado por una orden que indica si ésta ha terminado con éxito o no. El número se almacena en RAM por el MS-DOS y se puede leer con esta forma de la orden IF. Los programas que actualizan el nivel de error permiten que otros programas sepan si han terminado correctamente o no. Por ejemplo, un programa particular puede poner el nivel de error a 4, para indicar que no pudo encontrar cierto fichero. Desafortunadamente, no hay una estandarización que permita entender lo que significan los números de nivel de error. Sin embargo, cada orden que usa el nivel de error tiene un significado específico para cada número. Se puede determinar el nivel de error de la última orden ejecutada, resolver el problema y decidir qué orden del MS-DOS ejecutar a continuación, según el significado del nivel de error.

Puesto que muy pocas órdenes del MS-DOS actualizan el nivel de error, esta forma de la orden IF no es útil todavía. Futuras revisiones del MS-DOS y futuros paquetes de aplicación tendrán quizá más órdenes que actualicen el nivel de error.

Reglas

Si se usa la condición para comparar dos cadenas, las dos cadenas tienen que coincidir exactamente para que la condición sea verdad.

Por ejemplo, si una línea de su fichero batch es

IFSTRING.BAT

```
IF %3==MOVE COPY TOTCOST.TXT B:
```

tiene que introducir «MOVE» para ejecutar la orden COPY. El MS-DOS no considerará que las cadenas coinciden si usted introduce «move» o «Move».

Si no se han introducido tantos argumentos como especifica el fichero batch, el MS-DOS dará un error.

Por ejemplo, si sólo da dos argumentos en la línea de orden, cuando ejecuta la línea del ejemplo previo, el MS-DOS visualizará «Syntax error» (Error de sintaxis) y continuará.

Advertencias y errores frecuentes

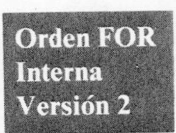
No se pueden usar nombres de caminos con la condición EXIST.

Sintaxis

La sintaxis de la orden IF es

IF [NOT] *condición orden*

Las tres *condiciones* disponibles se describieron en la tabla 3-8. La *orden* es cualquier orden (como por ejemplo, una orden del MS-DOS, un programa de aplicación o una orden de fichero batch). NOT niega la *condición*.



Orden FOR
Interna
Versión 2

La orden FOR permite repetir una orden para muchos argumentos diferentes. También permite repetir una orden para todos los ficheros que se corresponden con un nombre de fichero que contiene caracteres comodín.

Usos más comunes

La orden FOR se usa para ejecutar una orden repetidamente sobre una lista de argumentos. La primera sintaxis usual de la orden FOR es

FOR %%*letra* IN (*lista*) DO *orden*

La *letra* es cualquier letra del alfabeto; es igual que un rótulo para una celda de RAM. Cada elemento de la *lista* se copia en ese lugar de RAM, uno

cada vez, y la *orden* se ejecuta utilizando ese elemento de la lista. A continuación se copia el siguiente elemento y se repite el procedimiento. Este concepto se explica mejor con un ejemplo.

Por ejemplo, para obtener el directorio de A:, B:, C: y D:, podría incluir las siguientes órdenes en un fichero batch:

HARDWAY.BAT

```
DIR A:
DIR B:
DIR C:
DIR D:
```

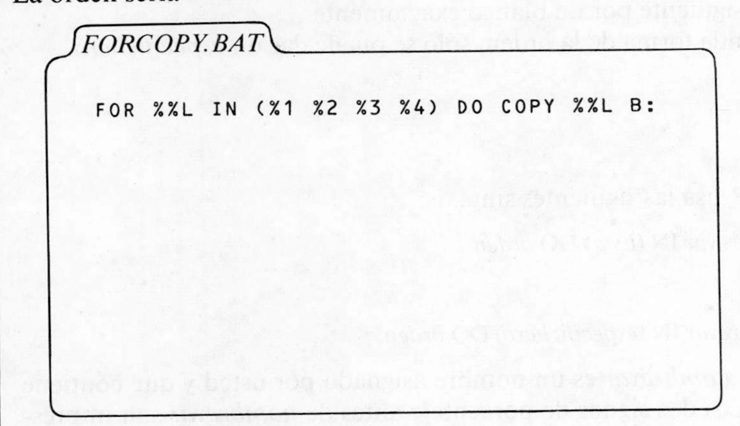
Una forma más fácil de realizar este fichero batch sería usar la orden FOR. En este caso, la orden sería

EASYWAY.BAT

```
FOR %%A IN (A: B: C: D:) DO DIR %%A
```

Esta orden le dice al MS-DOS que ejecute una orden DIR en los discos A:, B:, C: y D:.

También se puede usar esta forma de la orden FOR para copiar todos los ficheros de la línea de orden de un fichero batch a otro disco en la unidad B:. La orden sería



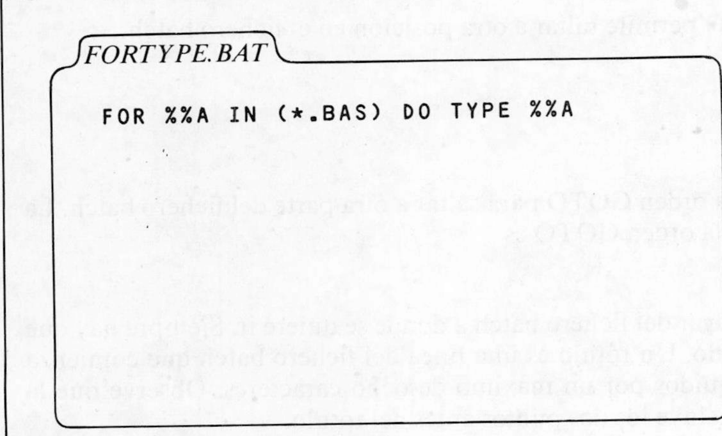
Esta orden copiará en B: cuatro ficheros que usted nombra en la línea de la orden.

La segunda forma de sintaxis de la orden FOR es

FOR %%*letra* IN (*especifichero*) DO *orden*

En esta forma, *especifichero* es una especificación de fichero que contiene caracteres comodín. En este caso, el MS-DOS realiza la *orden* con cada fichero que encaja con la especificación. Esta forma sólo usa un argumento en el paréntesis.

Por ejemplo, para realizar una orden TYPE con cada fichero de un disco, que tenga una extensión BAS, dé la orden



Reglas

Si usa la primera forma de la orden, cada elemento de la lista tiene que estar separado del siguiente por un blanco exactamente.

Si usa la segunda forma de la orden, sólo se puede dar un *especifichero.*

Sintaxis

La orden FOR usa las siguientes sintaxis:

FOR %%*nombrevar* IN (*lista*) DO *orden*

o bien

FOR %%*nombrevar* IN (*especifichero*) DO *orden*

El argumento *nombrevar* es un nombre asignado por usted y que contiene una única letra. Los dos signos de porcentaje antes de *nombrevar* son imprescindibles.

En la primera forma, *lista* es una lista de argumentos separados por un espacio. En la segunda forma, *especifichero* es una especificación de fichero que contiene caracteres comodín; la orden FOR lo trata como si fuera una lista de ficheros que emparejan con *especifichero.* La *orden* se ejecuta con cada elemento de la *lista.*

Orden GOTO
Interna
Versión 2

La orden GOTO permite saltar a otra posición en el fichero batch.

Usos más comunes

Se puede usar la orden GOTO para saltar a otra parte del fichero batch. La sintaxis general de la orden GOTO es

GOTO *rótulo*

El *rótulo* es el lugar del fichero batch a donde se quiere ir. Siempre hay que saltar hasta un rótulo. Un rótulo es una línea del fichero batch que comienza con dos puntos seguidos por un máximo de ocho caracteres. Observe que la orden GOTO no incluye los dos puntos antes del rótulo.

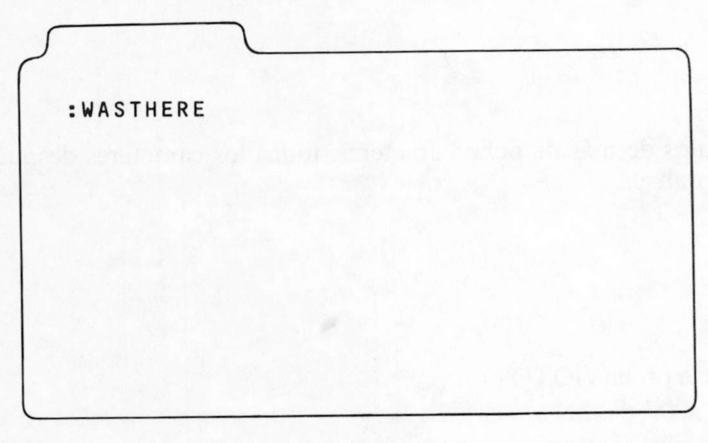
Un rótulo es igual que un poste indicador que le dice al procesador batch dónde comenzar de nuevo después de una orden GOTO. Cuando se ejecuta un fichero batch, los rótulos no se visualizan en la pantalla.

Por ejemplo, puede que usted quiera escribir un fichero batch que le indique si un fichero existe o no. El siguiente fichero, IFGOTO.BAT muestra cómo hacerlo usando la orden GOTO junto con la orden IF en un fichero batch.

IFGOTO.BAT

```
IF EXIST %1 GOTO WASTHERE
ECHO The file %1 was not there
GOTO DONE
:WASTHERE
ECHO The file %1 was there
:DONE
```

Esta orden batch visualiza un mensaje relativo a la existencia del fichero nombrado en el primer argumento de la orden. La primera línea comprueba si el fichero existe. Si existe, el procesador batch saltará a la línea



Si no existe, el procesador batch continúa con la siguiente línea (la primera orden ECHO).

También se puede saltar a un rótulo que está antes de la orden GOTO. Esto hace que el programa se ejecute en un ciclo sin fin. Esto puede ser útil si se quiere realizar la misma tarea una y otra vez. Para salir del ciclo sin fin, simplemente pulse CONTROL-C mientras se ejecuta el fichero batch.

Por ejemplo, un fichero batch que permita ejecutar la orden CHKDSK con tantos disquetes como se quiera podría ser algo como esto:

GOTOLOOP.BAT

```
ECHO OFF
:LOOP
ECHO Insert a disk in C:
PAUSE
ECHO Testing...
CHKDSK C:
GOTO LOOP
```

El fichero GOTOLOOP.BAT continúa ejecutando la orden CHKDSK hasta que el usuario pulse CONTROL-C.

Reglas

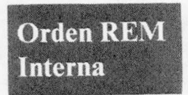
Si se dan rótulos de más de ocho caracteres, todos los caracteres después del octavo se ignoran.

Sintaxis

La sintaxis de la orden GOTO es

GOTO *rótulo*

El *rótulo* es la línea del fichero batch a la que se quiere ir.



La orden REM se usa para añadir comentarios a un fichero batch. No produce ningún resultado.

Usos más comunes

La orden REM se usa para poner comentarios en un fichero batch. Esta orden de puede usar para describir los contenidos de un fichero batch. Cuando escribe un fichero batch, usted mismo puede querer dejar notas de lo que hace dicho fichero.

Por ejemplo, el siguiente fichero REMARK.BAT incluye comentarios en las órdenes REM:

REMARK.BAT

```
ECHO OFF
REM         This file was written by Les
REM         Sanders to test diskettes
:LOOP
ECHO Insert a disk in C:
PAUSE
ECHO Testing...
CHKDSK C:
GOTO LOOP
REM          This GOTO causes an endless loop
```

Recuerde que los comentarios REM no hacen nada. Si ECHO está en ON, se visualizarán en la pantalla. Si ECHO se pone en OFF, como en este fichero batch, el mensaje REM sólo será visible visualizando el fichero REMARK.BAT.

Siempre es una buena idea poner al menos una orden REM en un fichero para recordar por qué se escribió el fichero.

Sintaxis

La sintaxis de la orden REM es

REM *comentario*

El argumento *comentario* es cualquier texto. La orden REM se puede colocar en cualquier parte de un fichero batch.

Orden SHIFT
Interna
Versión 2

La orden SHIFT desplaza todos los argumentos en un número hacia abajo, permitiendo el acceso a argumentos posteriores al %9. Si no se usa la orden SHIFT, el MS-DOS limita a nueve el número de argumentos en la línea de la orden de un fichero batch. La orden SHIFT se usa muy raramente.

Usos más comunes

Para lo único que se utiliza la orden SHIFT es para tener acceso a más de nueve argumentos de la línea de la orden; es improbable que usted necesite dar más de nueve argumentos en un fichero batch.

La orden SHIFT cambia el argumento %0 del fichero batch para leer %1 de la línea de la orden, %1 lo cambia a %2 y así sucesivamente hasta cambiar %9 a %10, permitiendo así el acceso a un argumento más. Después de una orden SHIFT, no se puede ver el contenido original del argumento %0. Hay que dar una orden SHIFT por cada argumento que se quiere usar, después del noveno. De este modo, si quiere usar 12 argumentos, tendrá que incluir tres órdenes SHIFT en su fichero batch.

Lo siguiente es un ejemplo simple de un fichero batch que usa la orden SHIFT:

SHIFTEX.BAT

```
ECHO OFF
ECHO %1 %2
ECHO %2 %3
SHIFT
ECHO %1 %2
```

Cuando se ejecuta este fichero batch, en la pantalla aparece algo como esto:

```
A>SHIFTEX AAA BBB CCC DDD

A>ECHO OFF
AAA BBB
BBB CCC
BBB CCC
```

Sintaxis

La sintaxis de la orden SHIFT es

SHIFT

SHIFT desplaza automáticamente todos los argumentos % de un fichero batch un número hacia abajo.

Orden COMMAND
Externa
Versión 2

He aquí un interesante trabalenguas: «the COMMAND command is MS-DOS's command processor» (la orden COMMAND es el procesador de órdenes del MS-DOS). Se puede usar el argumento /C de la orden COMMAND para ejecutar un fichero batch desde dentro de un fichero batch.

Usos más comunes

En la discusión de los ficheros batch en el capítulo 2 se describió cómo se podía enlazar un fichero batch con otro, dando el nombre del segundo fichero batch. Se explicó allí que usted podía enlazar por el final de un fichero batch, porque el MS-DOS casi nunca vuelve al fichero batch inicial. La orden COMMAND es una excepción a esta generalización: permite pasar de un fichero a otro y a continuación volver al fichero inicial, cuando se termina el segundo. La sintaxis usual para hacer esto es

COMMAND /C *orden*

Hay otras formas de usar la orden COMMAND, pero el argumento /C es la forma más importante y la única descrita aquí.

Compare los dos ejemplos que vienen a continuación. Ilustran la diferencia entre un fichero que no incluye una orden COMMAND y otro que sí lo hace.

Este ejemplo sólo pretende mostrar por qué la orden COMMAND es importante; no sirve para otra cosa. Sean los dos ficheros batch:

TEST1.BAT

```
ECHO Now in TEST1
TEST2
ECHO Still in TEST1
```

TEST2.BAT

```
ECHO Now in TEST2
```

Cuando se ejecuta TEST1, aparece lo siguiente en la pantalla:

```
A>TEST1

A>ECHO Now in TEST1
Now in TEST1
A>TEST2

A>ECHO Now in TEST2
Now in TEST2

A>
```

Desde que usted ve el cursor aquí, puede afirmar que el MS-DOS ha terminado. Esto significa que, después de ejecutar la única línea de TEST2, el MS-DOS no vuelve a TEST1, y la última línea de TEST1 no se ejecuta nunca. Esto es debido a que el MS-DOS enlaza con el segundo fichero batch y no recuerda adónde volver.

Puede que usted encuentre conveniente ejecutar un fichero batch a partir de otro y, cuando se acabe el segundo fichero, continuar procesando el fichero inicial. La orden COMMAND le permite hacerlo.

Puede cambiar el fichero TEST1 para hacer que el MS-DOS vuelva a él después de ejecutar TEST2:

TEST1.BAT

```
ECHO Now in TEST1
COMMAND /C TEST2
ECHO Still in TEST1
```

TEST2.BAT

```
ECHO Now in TEST2
```

Observe cómo la línea que antes era

```
TEST2
```

ahora es

```
COMMAND /C TEST2
```

La opción /C le dice a la orden COMMAND que el otro argumento es una orden que debe ser ejecutada. Esta línea en un fichero batch hace que ahora el MS-DOS ejecute el segundo fichero batch y que vuelva al primero:

```
A>TEST1

A>ECHO Now in TEST1
Now in TEST1

A>COMMAND /C TEST2

A>ECHO Now in TEST2
Now in TEST2

A>ECHO Still in TEST1
Still in TEST1

A>
```

Como puede ver, esto da a los ficheros batch del MS-DOS mucha mayor flexibilidad. Si está escribiendo un fichero batch grande, puede descomponerlo en ficheros más pequeños y usar la orden COMMAND para cada parte.

Advertencias y errores frecuentes

Tenga cuidado de no sobrecargar la memoria de su sistema. El fichero COMMAND.COM ocupa alrededor de 5K de RAM. Si usa la orden COMMAND con un programa que necesita la memoria que está libre normalmente en su sistema, puede que el programa se salga de la memoria. Sin embargo, esto ocurre muy raramente.

Si da la orden COMMAND sin el argumento /C, perderá una pequeña parte de RAM libre (realmente no hay razón para que usted lo haga). Para liberar esta memoria, dé la orden EXIT. La orden EXIT sólo es útil para este propósito, por ello no se describe en ninguna otra parte del libro.

Sintaxis

La sintaxis de la orden COMMAND es

COMMAND /C *orden*

El argumento /C le dice al MS-DOS que cargue temporalmente en memoria una copia del procesador de órdenes, ejecute la orden y, a continuación, elimine la copia del procesador de órdenes.

OTRAS ORDENES

Esta sección trata las siguientes órdenes:

FIND	*Busca texto en un fichero*
SORT	*Ordena un fichero texto*
CLS	*Borra la pantalla*
GRAPHICS	*Permite imprimir pantallas gráficas*

Las órdenes de esta sección no encajan en ninguno de los otros tipos de órdenes. FIND y SORT se usan a veces para manipular ficheros y son prácticas para realizar búsquedas u ordenaciones rápidas. La orden CLS es útil para borrar texto de la pantalla.

Orden FIND
Externa
Versión 2

La orden FIND encuentra todas las líneas de un fichero (o grupo de ficheros) que contienen una cadena determinada, y las visualiza en la pantalla. La

orden FIND utiliza entrada y salida redirigidas. Las entradas y salidas redirigidas se describen en el capítulo 4, pero no necesita saber cómo usarlas para usar la orden FIND.

Usos más comunes

Si quiere visualizar cada línea de un fichero que contenga una cadena en la que está interesado, use la siguiente sintaxis:

FIND «*cadena*» *fichero*

El argumento *cadena* tiene que estar entre comillas.

Por ejemplo, para buscar en el fichero DAILYRPT.TXT todas las líneas que contengan la expresión «well written», escriba la orden:

```
A>FIND "well written" DAILYRPT.TXT

---------- DAILYRPT.TXT
project was well written in a timely fashion. I do not feel
to the company. Another important job that can be well written
```

La primera línea es la forma que tiene la orden FIND de decirle el fichero con el que está trabajando. Las siguientes dos líneas son las líneas que ha encontrado en el fichero que contienen la cadena «well written».

La orden FIND visualiza cada línea que contiene la cadena que usted especifica; esto puede ser un poco desconcertante, puesto que no se puede ver la frase completa que contenía la expresión buscada. Desafortunadamente, FIND no sabe lo que son frases, sólo entiende de líneas y ficheros.

También se puede usar FIND con una lista de ficheros.

Por ejemplo, para encontrar todas las líneas que contienen la cadena «well written» tanto en el fichero DAILYRPT.TXT como en el fichero JEAN.RPT, introduzca:

```
A>FIND "well written" DAILYRPT.TXT JEAN.RPT

---------- DAILYRPT.TXT
project was well written in a timely fashion. I do not feel
to the company. Another important job that can be well written

---------- JEAN.RPT
I cannot find a well written sample of his writing that
```

El argumento /C indica a la orden FIND que sólo cuente las líneas, no que las visualice. El argumento /C, igual que todos los demás argumentos de

FIND, tiene que preceder a la cadena. Así, para el argumento /C la sintaxis usual es

FIND /C «*cadena*» *fichero*

Por ejemplo,

```
A>FIND /C "well written" DAILYRPT.TXT
---------- DAILYRPT.TXT: 2
```

El MS-DOS visualiza el número de veces que la cadena aparece en DAILYRPT.TXT.

Si quiere ver todas las líneas que no contienen una cierta cadena, use el argumento /V.

Por ejemplo,

```
A>FIND /V "well written" DAILYRPT.TXT
```

Como puede suponer, esto causará un enorme listado de líneas. Sin embargo, puede combinar argumentos de forma que pueda contar el número de líneas que no contienen una cierta cadena.

```
A>FIND /V/C "well written" DAILYRPT.TXT
---------- DAILYRPT.TXT: 189
```

El argumento /N le indica a la orden FIND que imprima el número de línea junto con ésta.

Por ejemplo:

```
A>FIND /N "well written" DAILYRPT.TXT
---------- DAILYRPT.TXT
[17]project was well written in a timely fashion. I do not feel
[131]to the company. Another important job that can be well written
```

El número de línea aparece al principio de cada línea que contiene la cadena.

Reglas

No se pueden usar caracteres comodín en los nombres de ficheros que se introduzcan. Hay que poner todos los argumentos que se usen, antes de la cadena y de los nombres de ficheros.

Si la cadena que quiere buscar tiene en su interior comillas ("), tiene que poner otras comillas delante de las primeras.

Por ejemplo, para buscar en el fichero DICKENS.TXT la expresión

```
home," he said.
```

escriba la orden:

```
A>FIND "home,"" he said." DICKENS.TXT
```

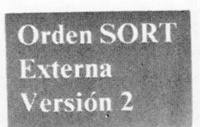
Sintaxis

La orden FIND usa la siguiente sintaxis:

FIND [/V] [/C] [/N] «*cadena*» [*especifichero*...]

La orden FIND busca la *cadena* en la lista de *especificheros* (la lista tiene espacios entre cada dos *especificheros*). La *cadena* tiene que ir entre comillas.

El argumento /V hace que FIND visualice cada línea que no contiene la *cadena.* El argumento /C indica que sólo se quiere visualizar el número total de líneas encontradas, y no las líneas en sí. El argumento /N hace que se incluya el número de línea entre corchetes ([ ]), al principio de la línea.



La orden SORT visualiza un fichero en una forma ordenada. La orden SORT también utiliza entrada y salida redirigidas, pero no se necesita saber cómo usar esto, para poder usar la orden SORT.

Usos más comunes

La orden SORT ordena un fichero según los valores ASCII de los caracteres del fichero. Esto significa que la mayoría de los signos de puntuación vendrán los primeros, seguidos por los números, las letras mayúsculas y las

minúsculas (la mayoría de los libros de programación contienen una lista ordenada de los códigos ASCII).

La forma simple de la orden SORT es

SORT <*fichero*

Observe que es necesario el signo menor que (<).

Por ejemplo, para ver el fichero INVITE.LST ordenado, escriba la orden

```
A>SORT <INVITE.LST
Gardner, Karen       x421
Hodges, Ralph        x120
Nance, Coleman       x0
Sanders, John        x425
Sanders, Susan       x419
d'Tang, Irene        x430
```

Observe que d'Tang se ha colocado al final porque comienza con una letra minúscula.

Si quiere ver el fichero en orden inverso, incluya el argumento / R. Todos los argumentos de la orden SORT tienen que preceder a los ficheros redirigidos. Así, para el argumento / R, la sintaxis usual es

SORT / R <*fichero*

Por ejemplo:

```
A>SORT /R <INVITE.LST
d'Tang, Irene        x430
Sanders, Susan       x419
Sanders, John        x425
Nance, Coleman       x0
Hodges, Ralph        x120
Gardner, Karen       x421
```

También se puede indicar a la orden SORT que comience a clasificar por otra columna distinta de la primera; para esto se usa el argumento / + con el número de columna.

Por ejemplo, para clasificar la lista por los números de teléfono (que empiezan en la columna 21 del fichero previo), escriba la orden:

```
A>SORT /+21 <INVITE.LST
Nance, Coleman       x0
Hodges, Ralph        x120
Sanders, Susan       x419
Gardner, Karen       x421
Sanders, John        x425
d'Tang, Irene        x430
```

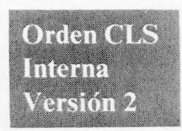
Sintaxis

La sintaxis de la orden SORT es

SORT [/R] [/+*n*] < *especifichero*

El signo menor que (<) es necesario para que la orden SORT funcione. El argumento *especifichero* representa al fichero que se quiere ordenar y visualizar.

El argumento /R hace que el fichero se visualice clasificado en orden inverso. El argumento /+*n* le indica a la orden SORT que haga la ordenación comenzando en la columna *n*.



La orden CLS borra la pantalla y visualiza la petición de orden del MS-DOS.

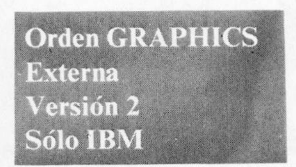
Usos más comunes

La orden CLS se usa para borrar la pantalla de la computadora. La razón principal para querer hacer esto suele ser el eliminar de la pantalla texto y gráficos, que podrían provocar confusión. Si se dejan los caracteres en el mismo sitio de la pantalla durante largo tiempo, accidentalmente se puede grabar la imagen de las letras en la pantalla. (Otra forma de evitar esto es bajar el brillo del monitor con el mando correspondiente).

Sintaxis

La sintaxis de la orden CLS es

CLS



La orden GRAPHICS permite copiar los gráficos de la pantalla a una impresora en el IBM PC. Sólo funciona con pantallas que tienen únicamente caracteres de texto normales o que están en uno de los modos gráficos.

Usos más comunes

La orden GRAPHICS hace que se cargue en RAM un programa de volcado de pantallas gráficas, que amplía el uso de la tecla PRTSC (PrintScreen) del IBM PC. Después de ejecutar la orden GRAPHICS, puede usar también PRTSC para imprimir pantallas desde modos gráficos. El manual del IBM PC-DOS da más detalles acerca de los modos gráficos y los procedimientos de volcado de pantallas.

Sintaxis

La sintasis de la orden GRAPHICS es

GRAPHICS

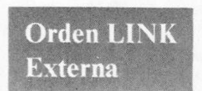
HERRAMIENTAS DE PROGRAMACION

Esta sección trata las siguientes órdenes:

LINK	*Combina ficheros objeto*
DEBUG	*Examina y cambia ficheros binarios*
EXE2BIN	*Convierte ficheros EXE a ficheros COM*

Las tres herramientas de programación, también llamadas utilidades (LINK, DEBUG y EXE2BIN) sólo son útiles para programadores avanzados. Incluso si usted aprende BASIC, no le serán útiles; realmente sólo sirven para la gente que programa en lenguaje ensamblador (aunque LINK a veces se usa en otros lenguajes de alto nivel). Es improbable que necesite aprender cómo usar estas órdenes (aunque la orden DEBUG puede ser útil).

Las siguientes descripciones de estas órdenes sólo pretenden servir como visión general. Las descripciones técnicas completas de las órdenes se pueden encontrar en la documentación que proporciona el fabricante de la computadora.



La orden LINK combina módulos objeto en programas que pueden ser ejecutados. Puede tener acceso a bibliotecas para ayudar a la definición de referencias externas de cada programa.

Sintaxis

La orden LINK usa los siguientes tipos de sintaxis:

LINK

o

LINK @ *especifichero*

o bien

LINK *listaobjeto, ficherosalida, ficheromapa, bibliotecas,*
[/DSALLOCATION] [/HIGH] [/LINE] [/MAP]
[/PAUSE] [/STACK:*n*]

La primera forma de la orden LINK hace que LINK le pregunte por todos los argumentos. Esto es útil si no quiere hacer una línea muy larga con la orden LINK. La segunda forma asume que usted ha creado un fichero que contiene las respuestas a las preguntas que LINK hace interactivamente. Es necesario el signo @ .

Los argumentos de la tercera forma de la orden LINK están definidos en la tabla 3-9. Cada uno de los argumentos puede ser abreviado a una letra.

Tabla 3-9. Argumentos de la orden LINK

Argumento	Significado
listaobjeto	Lista de ficheros objeto que van a ser enlazados, separados por signos más (+).
ficherosalida	Nombre del fichero EXE que va a ser producido.
ficheromapa	Nombre del fichero mapa que LINK va a producir.
bibliotecas	Lista de bibliotecas donde LINK debe buscar, separadas por signos más (+)
/DSALLOCATION	Indica que se quiere que la orden LINK cargue los datos DGROUP en el extremo de arriba del grupo.
/HIGH	Hace que la orden LINK coloque la imagen de ejecución lo más arriba posible en la RAM.
/LINE	Indica que se quiere que se incluyan los números de líneas en el fichero mapa.
/MAP	Hace que la orden LINK liste todos los símbolos globales en el fichero mapa.
/PAUSE	Dice a la orden LINK que visualice un mensaje antes de generar el fichero EXE, para que se puedan cambiar los discos.
/STACK*n*	Pone el tamaño de la pila a *n*, independientemente de su valor previo.

Orden DEBUG
Externa

La orden DEBUG se usa para probar programas y también para inspeccionar y modificar programas (así como otros ficheros binarios). La orden DEBUG es como un editor de líneas, salvo que permite cambiar los pasos y los datos de un programa, en vez de las letras y las palabras de un fichero texto.

Sintaxis

La sintaxis de la orden DEBUG es

DEBUG [*especifichero*] [*args*]

especifichero es el fichero que se quiere modificar o ejecutar; *args* son los argumentos que serán utilizados por una orden si usted está usando DEBUG en un fichero COM o EXE.

Orden EXE2BIN
Externa

La orden EXE2BIN convierte un fichero EXE en un fichero COM, con tal de que el programa EXE no tenga un segmento pila y sea más pequeño de 64K. Algunos fabricantes han cambiado el nombre de esta orden a LOCATE.

Sintaxis

La sintaxis de la orden EXE2BIN es

EXE2BIN *especifichero1* [*especifichero2*]

El argumento *especifichero1* es el fichero que se va a convertir. El argumento *especifichero2* es el nombre del fichero de salida. Si no se especifica un fichero de salida, la orden EXE2BIN usa el mismo nombre, con la extensión BIN.

4

USO AVANZADO DEL MS-DOS

Utilización de subdirectorios y caminos para organizar los ficheros
Entrada y salida redirigidas
Tubos (Pipes)
Edición de las líneas de órdenes del MS-DOS

Este capítulo explora tres conceptos que Microsoft introdujo con la versión 2 del MS-DOS: caminos, I/O redirigida y tubos (pipes). De estas tres características, disponibles sólo en la versión 2 del MS-DOS, la mayoría de los usuarios probablemente llegarán a la conclusión de que la utilización de caminos es la característica más útil, aunque la I/O redirigida puede ser más fácil de entender. Este capítulo también explica cómo usar teclas especiales del teclado para editar la línea de la orden, mientras se está escribiendo. Estas teclas son útiles independientemente de la versión del MS-DOS que posea.

Si no tiene la versión 2 del MS-DOS, debería por lo menos leer este capítulo rápidamente para que conozca las características disponibles en la versión 2. Si, posteriormente, oye hablar de alguna de las ideas presentadas aquí, estará familiarizado con los conceptos generales. Muchos fabricantes de computadoras le enviarán versiones más recientes del MS-DOS en cuanto estén disponibles, así que el conocer lo que está disponible le permitirá conocer lo que va a venir en el futuro.

UTILIZACION DE SUBDIRECTORIOS Y CAMINOS PARA ORGANIZAR LOS FICHEROS

Un directorio es una lista de los ficheros de un disco. Cuando se da la orden DIR, el MS-DOS visualiza esa lista de ficheros en la pantalla. Probablemente, usted piensa en las formas de subdividir el directorio en grupos de ficheros; en realidad, algunos de estos grupos pueden tener también subgrupos.

Agrupar los ficheros de esta forma sería lo mismo que construir un *directorio en forma de árbol.* Un directorio en forma de árbol es similar a un diagrama de organización a alto y bajo nivel. Cada parte del árbol de subdirectorios puede tener otros subdirectorios y ficheros por debajo. Por ejemplo, se pueden agrupar todos los ficheros texto de un disco en un subdirectorio con sus propios subdirectorios, y todos los ficheros de órdenes y programas en un directorio diferente.

La figura 4-1 muestra un dibujo de un árbol de subdirectorios. (En breve, se verá cómo construir este árbol en el disco). Esta figura muestra cómo se podrían organizar los directorios en un disco, y nos referiremos a ella frecuentemente. Cada caja contiene el nombre de un subdirectorio; los ficheros de cada subdirectorio se muestran en el lado derecho del subdirectorio. Como puede ver, algunos subdirectorios tienen subdirectorios adicionales por debajo, mientras que otros no; algunos tienen ficheros, mientras que otros sólo tienen subdirectorios.

Los subdirectorios de la figura 4-1 forman un dibujo semejante a las raíces de un árbol con el *directorio raíz* (directorio principal) en la parte de arriba. El subdirectorio que está inmediatamente encima de otro subdirectorio se llama *padre* y los que están justo debajo de él se llaman *hijos.* En la figura, STEIN y SMITH son hijos de CLIENTS.

Si tiene un gran número de ficheros en un disco duro, el poder separar los ficheros en subdirectorios le ayudará a organizar sus ficheros. Si no subdivide su directorio, cuando use la orden DIR, verá todos los ficheros sin orden aparente, a menudo llenando muchas pantallas. Si tiene un complicado sistema de directorios y subdirectorios, necesitará un *camino* que proporcione instrucciones para encontrar un directorio particular. Como veremos, los caminos permiten encontrar la vía adecuada en un directorio arborescente.

Cada directorio puede tener virtualmente tantos ficheros o subdirectorios como quepan en el disco, aunque en algunas implementaciones de la versión 2, el directorio raíz está limitado a 64 ficheros y subdirectorios. Sin embargo, esto no debe preocuparle. Demasiados subdirectorios pueden causar confusión. En la figura 4-1, la raíz tiene tres subdirectorios, el subdirectorio PROGRAMS tiene dos subdirectorios y el subdirectorio TEXT tiene tres subdirectorios.

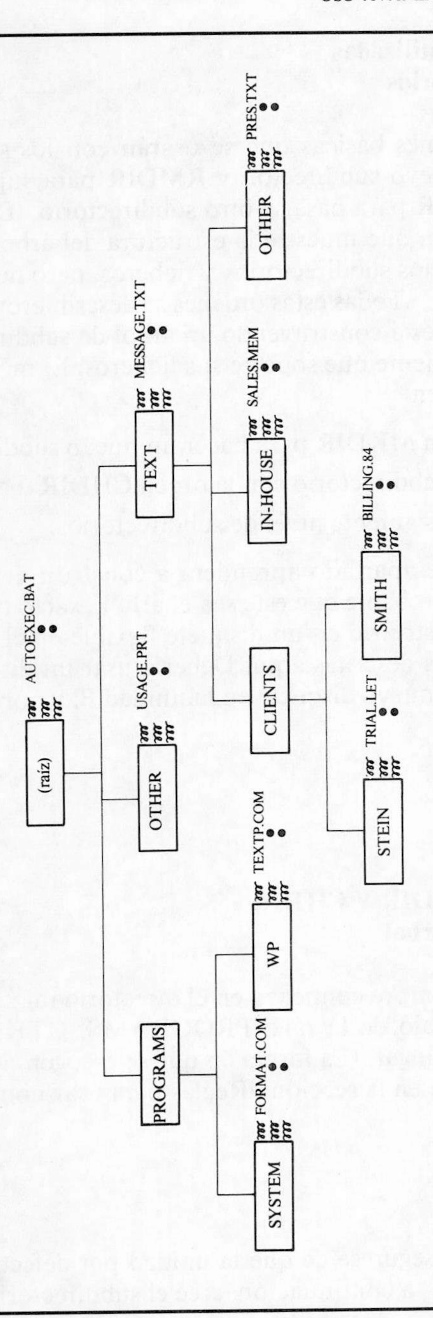


Figura 4-1. Un directorio en forma de árbol mostrando subdirectorios con ficheros

Ordenes básicas utilizadas con los subdirectorios

Hay tres órdenes básicas que se usarán con los subdirectorios: MKDIR para hacer un nuevo subdirectorio, RMDIR para suprimir un subdirectorio existente y CHDIR para pasar a otro subdirectorio. (Desafortunadamente, no hay ninguna orden que muestre la estructura del árbol de un disco. La orden TREE lista todos los subdirectorios y ficheros, pero no los pone en un gráfico como la figura 4-1). Todas estas órdenes se describieron en el capítulo 3.

Recuerde que está construyendo un árbol de subdirectorios para crear una estructura conveniente que soporte sus ficheros. El método general para crear un árbol consiste en

- Usar la orden MKDIR para hacer un nuevo subdirectorio
- Pasar a ese subdirectorio con la orden CHDIR
- Construir el siguiente nivel de subdirectorios.

En el siguiente apartado aprenderá a construir el árbol de la figura 4-1. Puesto que es improbable que este sea el árbol exacto que usted necesita en su disco, debería construirlo en un disquete flexible en el que pueda experimentar con las órdenes descritas aquí. Debería usar un disquete que no contenga ficheros. Ponga el nuevo disquete en la unidad B: y ponga B: como unidad por defecto.

Utilización de MKDIR Y CHDIR para construir un árbol

El MS-DOS siempre comienza en el directorio raíz. Puede ver que hay tres subdirectorios debajo de la raíz: PROGRAMS, OTHER y TEXT. Estos se crearán en primer lugar. (La forma en que se escogen nombres para los subdirectorios se explica en la sección «Reglas para usar nombres de subdirectorios y de caminos»).

Para empezar, asegúrese de que la unidad por defecto es la B: (la petición de orden será B>) y, a continuación, cree el subdirectorio PROGRAMS:

```
B>MKDIR PROGRAMS
```

La orden MKDIR no le pasa a usted a este nuevo subdirectorio; usted está

todavía en la raíz. Ahora cree los subdirectorios OTHER y TEXT:

```
B>MKDIR OTHER

B>MKDIR TEXT
```

Puede usar la orden DIR para verificar que ha construido estos subdirectorios. La salida de la orden DIR está descrita con mayor detalle en la sección «Cómo entender los listados de las órdenes DIR y TREE».

Una vez que ha construido estos subdirectorios, puede pasar a ellos con la orden CHDIR y meterles ficheros, o bien construirles subdirectorios por debajo. (Naturalmente, también puede meter ficheros en el directorio raíz). Para hacer esto, utilice la orden CHDIR, que también se mencionó en el capítulo 3.

Utilice la orden CHDIR para pasar al subdirectorio PROGRAMS y, a continuación, añada los subdirectorios SYSTEM y WP por debajo de él:

```
B>CHDIR PROGRAMS

B>MKDIR SYSTEM

B>MKDIR WP
```

Para meter ficheros en un subdirectorio, simplemente pase a ese subdirectorio y utilice la orden COPY, igual que se hace normalmente.

Por ejemplo, asuma que el disco que contiene su procesador de texto está en la unidad A: y quiere copiar todos los ficheros de dicha unidad en el subdirectorio WP. Daría las órdenes

```
B>CHDIR WP

B>COPY A:*.*
```

Para volver al directorio raíz y construir el resto del árbol, necesita dar la orden

```
B>CHDIR \
```

La barra hacia atrás (\ ) es un nombre especial que el MS-DOS utiliza para la raíz; su uso se explica en la siguiente sección. Una vez que está de vuelta en la parte superior, puede acabar de crear el árbol.

Para terminar de construir el árbol, asegúrese de que está en el directorio raíz. (Si no lo está, dé la orden CHDIR \ que acabamos de describir. A continuación, escriba las siguientes órdenes:

```
B>CHDIR TEXT
B>MKDIR CLIENTS
B>MKDIR IN-HOUSE
B>MKDIR OTHER
B>CHDIR CLIENTS
B>MKDIR STEIN
B>MKDIR SMITH
B>CHDIR \
```

Ahora usted ha creado el árbol entero. Puede usar la orden CHDIR para bajar de nivel y, entonces, usar COPY para meter ficheros en cada uno de los subdirectorios. Cuando se mueva por el árbol, la única forma de moverse hacia arriba es con la orden CHDIR \ que le lleva a la raíz y, a continuación, descender a donde quiera ir. Más adelante aprenderá a usar los caminos para que no necesite seguir volviendo a la raíz.

Recuerde que cada disco puede tener un conjunto diferente de subdirectorios, de modo que el árbol del disco A: puede ser muy diferente del árbol del disco B:. Generalmente, sólo tendrá que recordar la forma de uno o dos árboles diferentes. Si no usa los subdirectorios del MS-DOS, todos sus ficheros serán considerados como parte del directorio raíz de ese disco.

Cuando se usa la orden CHDIR para moverse por el árbol, realmente se está cambiando el *subdirectorio por defecto.* El subdirectorio por defecto es el subdirectorio en el que el MS-DOS busca los ficheros (a no ser que se le indique algo diferente). Por ejemplo, cuando se da la orden DIR desde dentro de un subdirectorio, sólo se ve la lista de los ficheros de ese subdirectorio. Cuando se da una orden externa, el MS-DOS sólo la buscará en el subdirectorio por defecto (a no ser que se haya dado la orden PATH). Si quiere ver cuál es el subdirectorio por defecto, utilice la orden CHDIR sin argumentos.

Reglas para usar nombres de subdirectorios y de caminos

Hay unas cuantas reglas a seguir cuando se definen subdirectorios. Cada subdirectorio tiene que tener un nombre. Igual que con los ficheros, es importante dar a los subdirectorios nombres descriptivos para hacerlos más fáciles de manejar. En la figura 4-1 algunos de los nombres de los subdirectorios son PROGRAMS, WP y SMITH. Las reglas para nombrar subdirectorios son exactamente las mismas que para nombrar ficheros (tratadas en el capítulo 2). Usualmente no se usa una extensión cuando se nombra un subdirectorio, aunque está permitido. Los ficheros que están dentro de los subdirectorios, naturalmente, pueden tener extensiones.

Aunque las reglas para nombrar los subdirectorios son las mismas que para nombrar ficheros, no se puede realizar el acceso a los ficheros de los subdirectorios de la misma forma que a los ficheros del directorio raíz. Esto es debido a que cada subdirectorio tiene un único *nombre de camino* que el MS-DOS utiliza para encontrar el subdirectorio. El camino describe cómo está enlazado el subdirectorio con la raíz. El directorio raíz no tiene nombre de camino; esto se indica con una barra hacia atrás (\ ). Cada nombre sucesivo a lo largo del camino se separa también con una barra hacia atrás. Por ejemplo, en la figura 4-1 el nombre del camino para el subdirectorio PROGRAMS es \ PROGRAMS, puesto que está conectado directamente a la raíz; el nombre del camino para el subdirectorio STEIN es \TEXT \CLIENTS \ STEIN.

Este método de dar nombres de caminos permite tener el mismo nombre de subdirectorio en diferentes lugares del árbol. Observe que hay dos subdirectorios llamados OTHER en la figura 4-1: uno es el hijo de la raíz, y otro es el hijo de TEXT. El MS-DOS puede distinguir la diferencia porque los nombres de los caminos son distintos (\OTHER y \TEXT \OTHER). Sin embargo, no se pueden tener dos subdirectorios con el mismo nombre conectados al mismo subdirectorio.

Para realizar el acceso a un fichero de un subdirectorio particular, simplemente añada otra barra hacia atrás al final del nombre del camino y el nombre del fichero que usted necesita.

Por ejemplo, si el fichero SALES.MEM estuviera en el subdirectorio \TEXT \IN-HOUSE, lo visualizaría en la pantalla con la siguiente orden:

```
B>TYPE \TEXT\IN-HOUSE\SALES.MEM
```

Esta orden tomará el fichero del disco por defecto.

Si necesita dar el nombre de una unidad en un especificador de fichero que tiene un nombre de camino, debe añadirlo al comienzo del nombre del camino.

Así, si el fichero previo estuviera en el disco A:, la nueva orden TYPE sería

```
B>TYPE A:\TEXT\IN-HOUSE\SALES.MEM
```

Asegúrese de que no pone un espacio entre el nombre de la unidad de disco y el comienzo del camino.

Se pueden usar nombres de caminos en la orden CHDIR para pasar directamente de un subdirectorio a otro. Esto hace que el moverse por el árbol sea mucho más fácil que tener siempre que volver a la raíz.

Por ejemplo, para pasar al subdirectorio SMITH de la figura 4-1, habría que dar la orden

```
B>CHDIR \TEXT\CLIENTS\SMITH
```

Observe que cuando se da la orden CHDIR no importa desde qué subdirectorio se empieza. La barra hacia atrás (\) al comienzo del nombre del camino le indica al MS-DOS que comience en el directorio raíz. Naturalmente, volver al directorio raíz desde cualquier subdirectorio es siempre fácil, puesto que tiene un nombre de camino muy corto:

```
B:CHDIR \
```

Cuando se usa la orden CHDIR, no siempre es necesario comenzar el nombre del camino en el directorio raíz. En realidad, usted puede pasar a un subdirectorio del directorio actual en el que esté. Recuerde que, en ejemplos previos, cuando daba un nombre de camino completo, incluía una barra hacia atrás para el directorio raíz (\TEXT \CLIENTS \SMITH). Se debe omitir esta barra hacia atrás en el nombre del camino, para pasar a un subdirectorio que es hijo del subdirectorio por defecto. Simplemente dígale a CHDIR el nombre de ese hijo, sin dar su nombre de camino entero.

Por ejemplo, para pasar de \TEXT a \TEXT \CLIENTS, daría la orden

```
B:CHDIR CLIENTS
```

Observe que el nombre completo del camino no era necesario en este ejemplo. Así es como se movía por el árbol en la primera parte de este capítulo.

Se puede usar el nombre del hijo del subdirectorio para pasar del padre al hijo. Simplemente ponga una barra hacia atrás después del nombre del hijo y continúe el nombre del camino.

Por ejemplo, si quisiera pasar al subdirectorio por defecto de \TEXT a \TEXT\CLIENTS\SMITH, daría la orden

```
B:CHDIR CLIENTS\SMITH
```

Un nombre de camino que muestra todo el camino desde la raíz se denomina *nombre de camino absoluto,* puesto que se está dando el nombre más completo posible. Cuando se da un nombre de camino basado en la posición actual, se denomina *nombre de camino relativo.* Un nombre de camino que empieza con una barra hacia atrás siempre significa que se parte del directorio raíz; un nombre de camino que comienza sin una barra hacia atrás siempre significa que se parte del subdirectorio actual.

Pasar a un subdirectorio hijo con un nombre de camino relativo puede causar alguna confusión, puesto que la orden es similar a cuando se pasa a un hijo de la raíz.

Por ejemplo, si está en el subdirectorio TEXT y quiere pasar al subdirectorio PROGRAMS, usted usa la barra hacia atrás:

```
B:CHDIR PROGRAMS
Invalid directory

B:CHDIR \PROGRAMS

B:
```

Para pasar al padre del subdirectorio por defecto, se usa un nombre de camino especial, constituido por dos puntos (..). A menudo se le llama *doble punto.* Este doble punto representa simplemente al padre del subdirectorio por defecto, independientemente de en qué parte del árbol se esté.

Por ejemplo, si su subdirectorio por defecto es WP y quiere pasar a su padre, el subdirectorio PROGRAMS, debe dar la orden

```
B:CHDIR ..
```

Pasar al subdirectorio padre con el doble punto es similar a pasar al subdirectorio hijo, porque se puede usar el doble punto como comienzo de un nombre de camino más largo. En realidad, el uso del doble punto es una forma corriente de pasar a un directorio del mismo nivel en el árbol que el subdirectorio actual.

Por ejemplo, para pasar de WP a SYSTEM, dé la orden

```
B:CHDIR ..\SYSTEM
```

También puede usar una serie de dobles puntos para moverse hacia arriba en el árbol. Para pasar de STEIN a TEXT, puede dar la orden

```
B:CHDIR ..\..
```

El uso de nombres de caminos en la orden MKDIR permite añadir subdirectorios en cualquier parte del árbol y no solamente en el subdirectorio actual. Simplemente utilice el nombre del camino completo en la orden MKDIR.

Por ejemplo, suponga que está en el subdirectorio \TEXT\IN-HOUSE y quiere añadir un subdirectorio a \PROGRAMS llamado GAMES. No necesita cambiar de subdirectorio con la orden CHDIR; simplemente escriba la orden:

```
B:MKDIR \PROGRAMS\GAMES
```

Esto especifica exactamente dónde quiere el nuevo subdirectorio.

Cómo entender los listados de las órdenes DIR y TREE

Ahora que sabe más de los subdirectorios, le será fácil entender el listado de la orden DIR que se obtiene cuando un disco tiene subdirectorios. Lo siguiente es un ejemplo del listado de DIR, tomado del subdirectorio \TEXT de la figura 4-1.

```
B>CD \TEXT

B>DIR

 Volume in drive B has no label
 Directory of  B:\text

.            <DIR>        2-11-84   10:54a
..           <DIR>        2-11-84   10:54a
CLIENTS      <DIR>        4-07-84    9:12p
IN-HOUSE     <DIR>        6-30-84    3:41a
OTHER        <DIR>        3-28-84    7:17p
MESSAGE  TXT          91 11-19-84    3:29p
        6 File(s)    303104 bytes free
```

Sólo hay un fichero en este subdirectorio, MESSAGE.TXT. La orden DIR no lista ficheros de otros subdirectorios, ni siquiera de los que están por debajo del actual.

La segunda línea da el nombre del subdirectorio al que corresponde el listado. Observe que hay cinco líneas que tienen <DIR>; estas líneas son entradas para subdirectorios, no ficheros. Probablemente reconozca el doble punto (..) como padre y CLIENTS, IN-HOUSE y OTHER como los tres hijos. El MS-DOS también incluye un directorio punto (.), que es el nombre para el subdirectorio actual; realmente no hay ningún uso para esta entrada.

Puede que se pregunte por qué la última línea informa de seis ficheros, cuando sólo puede ver uno que no es un directorio. Esto es debido a que el MS-DOS guarda toda la información interna relativa a subdirectorios padres y subdirectorios hijos como entradas de ficheros en cada directorio. De este modo, cuenta cada entrada —para subdirectorio hijos, así como para los subdirectorios punto y doble punto— como fichero. Estas entradas se ven cuando se usa la orden DIR; sin embargo, puesto que no se pueden tratar estas entradas como ficheros, no puede cambiarlas de nombre ni copiarlas.

El directorio raíz no tiene entradas de directorio punto ni doble punto:

```
B>CHDIR \

B>DIR

 Volume in drive B has no label
 Directory of  B:\

PROGRAMS     <DIR>        2-11-84   10:54a
OTHER        <DIR>        2-11-84   10:54a
TEXT         <DIR>        2-11-84   10:54a
AUTOEXEC BAT          91 11-19-83    3:29p
        4 File(s)    303104 bytes free
```

La orden TREE (que es una orden específica de IBM) puede ayudarle a imaginar la forma de un árbol sin tener que dar las órdenes CHDIR y DIR

para cada subdirectorio. He aquí la primera parte de la salida de la orden TREE para el árbol de la figura 4-1:

```
B>A:TREE /F

DIRECTORY PATH LISTING FOR VOLUME ???????????

Path: \PROGRAMS
Sub-directories:  SYSTEM
                  WP
Files:            None

Path: \PROGRAMS\SYSTEM
Sub-directories:  None
Files:            FORMAT  .COM

Path: \PROGRAMS\WP
Sub-directories:  None
Files:            TEXTP   .COM

Path: \OTHER
Sub-directories:  None
Files:            USAGE   .RPT

Path: \TEXT
Sub-directories:  CLIENTS
                  IN-HOUSE
                  OTHER
Files:            MESSAGE .TXT

           .
           .
           .
```

Recuerde del capítulo 3 que TREE es una orden externa. A diferencia de CHDIR y RMDIR, que son órdenes internas, tiene que dirigir al MS-DOS a la unidad que contiene el disco del sistema, antes de que pueda ejecutar la orden TREE. Sin el argumento /F, TREE listaría sólo los subdirectorios y no los ficheros.

Utilización de RMDIR para suprimir subdirectorios

Suprimir subdirectorios es tan fácil como añadirlos, salvo que usted debería primero estar seguro de que el subdirectorio está vacío. La orden que se usa para suprimir subdirectorios es la orden RMDIR.

Afortunadamente, el MS-DOS no le permitirá usar RMDIR en un subdirectorio que contenga ficheros; simplemente le dará un mensaje de error. El

MS-DOS no permite suprimir el subdirectorio actual, puesto que no sabría en qué subdirectorio colocarse. Esto sería análogo a si usted tirase de una alfombrilla situada bajo sus propios pies.

El uso de la orden RMDIR es igual que el de la orden MKDIR: se puede dar tanto un nombre de camino absoluto como relativo.

Recuerde que, previamente en este capítulo, añadió el subdirectorio GAMES al subdirectorio \PROGRAMS. Puede suprimir este subdirectorio de la misma forma en que lo creó. Asumiendo que está vacío, introduzca la orden

```
B>RMDIR \PROGRAMS\GAMES
```

Si un subdirectorio tiene ficheros cuando usted usa la orden RMDIR, obtendrá un mensaje de error algo vago:

```
B>RMDIR SMITH
Invalid path, not directory,
or directory not empty
```

(Camino no válido, no es un directorio o directorio no vacío). Realmente, este es el mismo mensaje de error que se obtiene cuando se escribe mal el nombre del subdirectorio que se quiere suprimir.

Una advertencia especial: si da el nombre de un subdirectorio a la orden ERASE en vez de un especificador de fichero, borrará todos los ficheros de ese subdirectorio después de preguntar «Are you sure?» (¿Está seguro?). Esto es inoportuno, porque usted puede tener ficheros y subdirectorios con nombres similares.

Por ejemplo, introduzca la orden

```
B>ERASE SMITH
Are you sure? Y
```

Esto borra todos los ficheros del subdirectorio SMITH, aunque el subdirectorio permanece en el disco. La única forma de suprimir un subdirectorio es con la orden RMDIR.

Cómo encontrar programas con la orden PATH

Ahora que sabe cómo usar los caminos, puede que se pregunte cómo informar al MS-DOS acerca de los ficheros que están en caminos diferentes. Realmente, informar al MS-DOS sobre diferentes caminos es muy fácil con la orden PATH; informar a otros programas acerca de distintos caminos es mucho más difícil.

Normalmente, el MS-DOS busca sólo en el directorio (o subdirectorio) por defecto, cuando se da una orden externa. Esta es la razón por la que tuvo que incluir la unidad A: en la línea de la orden, cuando usted introdujo la orden TREE en un ejemplo previo.

La orden PATH realiza una función similar a la de incluir el nombre de la unidad: da al MS-DOS un lugar en donde buscar las órdenes externas. La lista de PATH se puede considerar como una lista de búsqueda ampliada, porque da al MS-DOS una lista de lugares donde buscar las órdenes. Esta es una ventaja agradable de tener caminos: puede decirle al MS-DOS que busque en muchos lugares diferentes del disco las órdenes que usted da. Es bastante fácil decirle al MS-DOS que usted ha almacenado sus programas en muchos caminos diferentes, e incluso en caminos de distintos discos.

Pero, ¿por qué querría usted guardar programas en diferentes subdirectorios? Puede que haya observado que el subdirectorio PROGRAMS de la figura 4-1 sería un sitio lógico para colocar los programas. Sus dos hijos, SYSTEM y WP, proporcionan un método fácil para separar las docenas de programas que usted tenga: SYSTEM podría contener todas las órdenes externas del MS-DOS y WP contendría su programa de procesamiento de texto. Naturalmente, mientras trabaja, usted quiere que todos los programas de estos dos directorios estén disponibles, independientemente de en cual de los subdirectorios del árbol se encuentre.

El orden de búsqueda del MS-DOS (que se fija con la orden PATH) hace más fácil la gestión de los ficheros de programas. La descripción de la orden PATH en el capítulo 3 explicaba que el argumento de la orden es una lista de nombres de subdirectorios separados por puntos y comas. Recuerde que independientemente de la lista de subdirectorios que especifique en la orden PATH, el MS-DOS siempre mirará primeramente en el subdirectorio actual.

Por ejemplo, puede dar al MS-DOS el siguiente orden de búsqueda:

- Busca en el subdirectorio por defecto
- A continuación busca en \PROGRAMS\SYSTEM
- A continuación busca en \PROGRAMS\WP.

Para hacerlo, daría la orden

```
B>PATH \PROGRAMS\SYSTEM;\PROGRAMS\WP
```

Observe que la unidad por defecto aquí es B:.

Es importante recordar que el MS-DOS necesita saber en qué disco debe buscar las órdenes. Si no se le dice, como en el último ejemplo, sólo mirará en el disco por defecto. No hay ningún mal en incluir explícitamente el nombre del disco con cada nombre de camino en la orden PATH.

Si dio la orden PATH (del ejemplo anterior) y el disco por defecto era A:, el MS-DOS buscará los caminos \PROGRAMS\SYSTEM y \PROGRAMS\WP en A:, que no es lo que usted quiere. En su lugar, debía haber dado la orden

```
B>PATH B:\PROGRAMS\SYSTEM;B:\PROGRAMS\WP
```

Esto asegura que el MS-DOS siempre buscará en el disco correcto. Si omite el nombre de la unidad en alguno de los caminos, el MS-DOS sólo buscará esos caminos en la unidad por defecto.

Cómo encontrar ficheros con nombres de caminos

Siempre que usted dé un nombre de camino explícito, el MS-DOS no tendrá problemas para encontrar ficheros en cualquier subdirectorio. La orden PATH sólo afecta a cómo el MS-DOS busca las órdenes externas. La orden PATH no afecta a cómo el MS-DOS busca otros ficheros.

En el ejemplo previo, el MS-DOS sólo encontrará ficheros de programas en los dos subdirectorios que usted especificó en la línea de la orden. Así, si su subdirectorio por defecto es \OTHER y el fichero que necesita, SALES.MEM, está en el subdirectorio \TEXT\IN-HOUSE, obtendrá un mensaje de error cuando dé la orden

```
B>TYPE SALES.MEM
File not found
```

El MS-DOS sólo encontrará este fichero si hay un fichero llamado SALES.MEM en el subdirectorio OTHER (el subdirectorio por defecto).

Para decirles a las órdenes del MS-DOS cómo encontrar otros ficheros, escriba el nombre del camino junto con el nombre del fichero.

La orden TYPE correcta en el ejemplo anterior es

```
B>TYPE \TEXT\IN-HOUSE\SALES.MEM
```

Las dos órdenes MS-DOS que no pueden buscar en otros subdirectorios son las órdenes batch FOR y EXIST. (No está claro por qué Microsoft deja éstas fuera). Esto significa que órdenes como COPY, ERASE y RENAME pueden encontrar ficheros, estén o no en el camino por defecto.

Desafortunadamente, alrededor de la mitad de los programas de aplicación disponibles para el MS-DOS no funcionan con ficheros que no están en el subdirectorio por defecto. Por ejemplo, si le da a un programa procesador de texto un nombre de camino para un fichero que quiere editar, puede que no sea capaz de entender lo que usted quiere. Esta situación lamentable se debe en su mayor parte a la negligencia de la gente que escribe y mantiene programas de aplicación. Muchos de estos programas fueron escritos primeramente para la versión 1 del MS-DOS, que no incluía caminos. Cuando aparecieron versiones más recientes, los programas no se actualizaron porque las compañías creyeron que era demasiado caro (o demasiado difícil) hacer los cambios necesarios para incluir nombres de caminos. Como resultado, hay poco que pueda hacer si tiene un programa de aplicación que no usa subdirectorios.

La única solución es poner el programa de aplicación en el mismo subdirectorio con todos los ficheros que usa. Esto significa que usted puede que tenga muchas copias idénticas de ficheros y programas en sus discos. Si esta restricción le obliga a copiar ficheros de un subdirectorio a otro para ejecutar el programa, puede (y debe) quejarse al fabricante de software por la falta de compatibilidad con el MS-DOS versión 2.

ENTRADA Y SALIDA REDIRIGIDAS

Otra característica que Microsoft añadió a la versión 2 del MS-DOS es la *entrada y salida redirigidas* o *I/O redirigida.* La I/O redirigida es la capacidad de desviar la entrada o salida de un programa de un dispositivo a otro. Con la I/O redirigida, usted puede hacer que un programa, que normalmente lee in-

formación del teclado, la lea de un fichero, o puede decirle a un programa, que normalmente escribe información en la pantalla, que la escriba en un fichero. La I/O redirigida puede ahorrarle una gran cantidad de escritura y de observación de la pantalla.

Se redirige la entrada cuando se le dice al MS-DOS que cada vez que un programa esté esperando una entrada por el teclado, el MS-DOS debe en su lugar darle la entrada desde un fichero. La salida redirigida, por otra parte, es cuando se le dice al MS-DOS que grabe un carácter en un fichero, cada vez que el programa quiera escribir un carácter en la pantalla. La figura 4-2 muestra cómo son la entrada y salida redirigidas.

Puede que haya observado que tres de las órdenes del capítulo 3 (MORE, SORT y FIND) necesitaban un signo de menor que (<) antes del nombre de fichero. Este signo añadido al argumento le dice al MS-DOS que estas órdenes están usando entrada redirigida (un signo de mayor que (>) indica salida redirigida). También se puede redirigir la entrada y la salida hacia un dispositivo. Simplemente, se da el nombre del dispositivo en lugar de un nombre de fichero, en la línea de la orden.

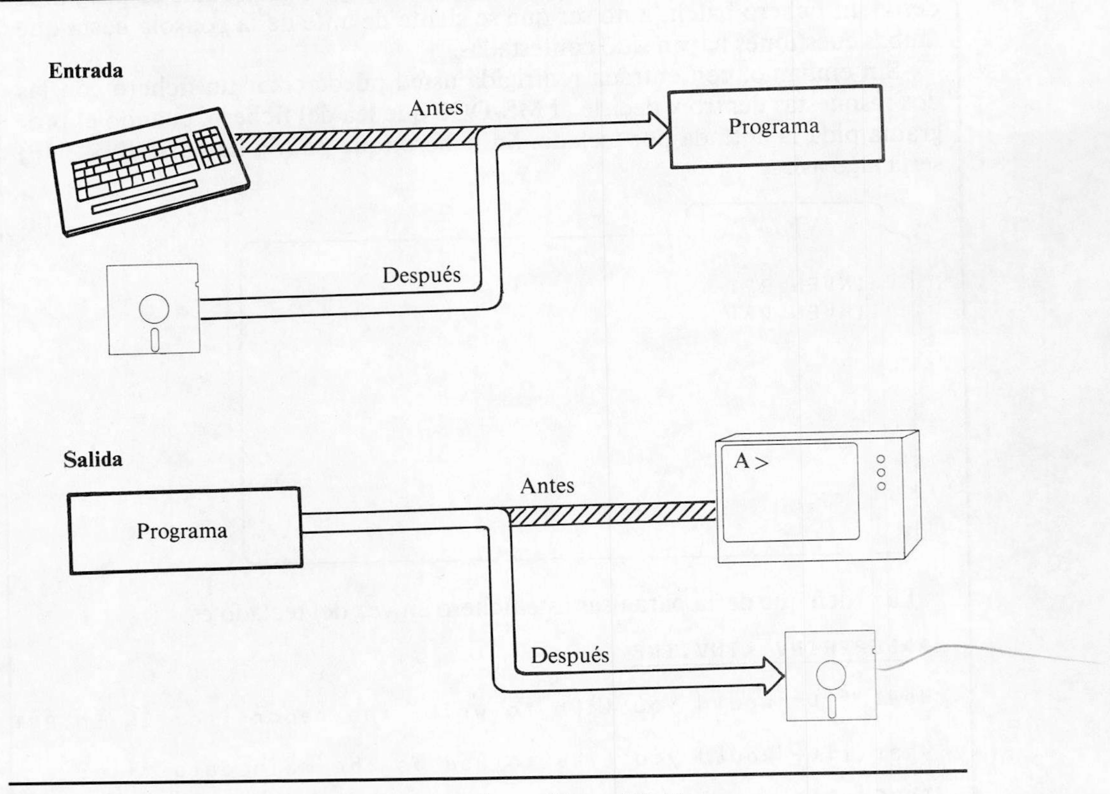


Figura 4-2. Entrada y salida redirigidas

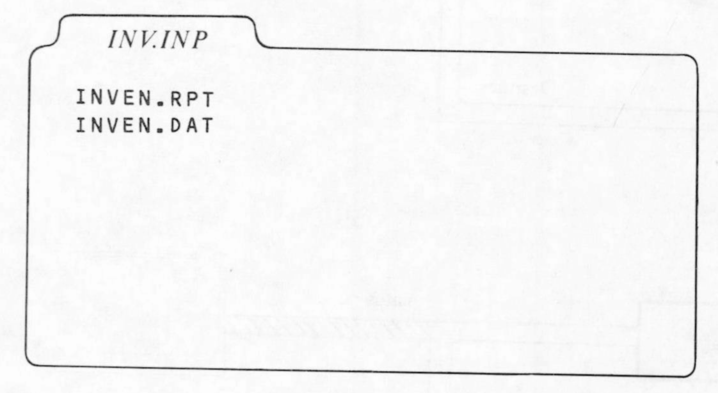
Entrada redirigida

La entrada redirigida puede ahorrarle mucho tiempo si usted normalmente tiene que introducir la misma información cada vez que ejecuta un programa. Utilizando entrada redirigida, puede poner sus respuestas usuales en un fichero y decirle al MS-DOS que lea del fichero en vez del teclado.

Para decirle al MS-DOS de qué fichero quiere leer, ponga un signo de menor que (<) delante del nombre del fichero. El signo de menor que se eligió para indicar que la entrada «viene desde» el fichero, puesto que se parece a una flecha que apunta desde el fichero hacia el nombre de la orden. (Este es el mismo método utilizado en otros sistemas operativos como el UNIX).

Suponga que tiene un programa generador de informes llamado SUPERINV que le hace dos preguntas: en qué fichero escribir el informe y qué fichero usar como base de datos. Cada vez que usted ejecuta el programa, siempre introduce las dos mismas respuestas INVEN.RPT e INVEN.DAT. Naturalmente, puesto que el programa le hace estas preguntas y espera que usted escriba las respuestas en el teclado, no se puede ejecutar el programa desde un fichero batch, a no ser que se siente delante de la consola hasta que ambas cuestiones hayan sido contestadas.

Sin embargo, con entrada redirigida usted puede crear un fichero con las dos respuestas dentro y decirle al MS-DOS que lea del fichero, cuando el programa pida la entrada por teclado. Su fichero (que podría llamarse INV.INP) sería algo así:



La orden que daría para usar este fichero en vez del teclado es

```
A>SUPERINV <INV.INP
What file would you like to write the report to: INVEN.RPT
What file would you like to use as the main data file:
INVEN.DAT
```

Observe que el MS-DOS escribe las respuestas por usted. Ahora puede incorporar esta orden en un fichero batch y el programa no volverá a necesitar que usted utilice el teclado mientras se ejecuta.

Hay algunos inconvenientes al utilizar entrada redirigida que requieren alguna preparación extra. Tiene que estar seguro de que ha contestado todas las preguntas que le hace su programa cada vez que le pide información. Si le hace una pregunta que usted no esperaba, los resultados pueden ser desastrosos. Si hay una línea en el fichero de entrada que estaba prevista como respuesta a una pregunta particular, pero se hace una pregunta diferente, el programa obtendrá la respuesta incorrecta. Por ejemplo, si usted había puesto «SI» en el fichero de entrada porque pensaba que el programa le iba a preguntar «¿Quiere comenzar ahora?», pero en vez de esto le pregunta «¿Quiere borrar el fichero?», podría perder información valiosa.

Otro inconveniente es que una vez que se ha redirigido la entrada para que venga desde un fichero, no se pueden ya introducir datos por el teclado. Si no hay más respuestas en el fichero de entrada y el programa hace otra pregunta, esperará indefinidamente a recibir la entrada y habrá que volver a hacer la carga inicial de la computadora. Este es un inconveniente muy serio para usar la entrada redirigida: si la utiliza para contestar el primer grupo de preguntas desde el fichero, todas las respuestas que el programa pida tienen que ser contestadas desde el fichero hasta que el programa termine.

Estas consideraciones limitan la utilidad de la entrada redirigida. Sin embargo, teniendo cuidado muchos programas simples pueden funcionar con entrada redirigida.

Salida redirigida

La razón más corriente para utilizar la salida redirigida es poder estudiar posteriormente la información producida por un programa. Otro uso de la salida redirigida es conservar la salida de un programa en un fichero que pueda ser editado posteriormente con un procesador de texto. En realidad, la mayoría de los ejemplos de este libro se guardaron de esta forma.

Por ejemplo, imagine que tiene que estudiar los datos producidos por un programa que escribe una enorme cantidad de información en la pantalla. Normalmente, tendría que mirar la pantalla y guardar toda la información escribiéndola manualmente. Esto le llevaría mucho tiempo. En cambio, usted puede redirigir la salida hacia un fichero que podrá examinar más tarde.

Para decirle al MS-DOS que todo lo que usualmente aparece en la pantalla debe ir a un fichero, se utiliza el signo de mayor que (>) y el nombre del fichero. El signo de mayor que indica que «va hacia»; puede recordarlo porque se parece a una flecha apuntando hacia el nombre del fichero.

La salida redirigida es útil para algunas de las órdenes del MS-DOS que dan voluminosa información. Por ejemplo, la orden CHKDSK con el argu-

mento /V usualmente visualiza muchas pantallas de información cuando se ejecuta. En vez de tener que mirar cuidadosamente toda esta información, puede enviarla a un fichero para examinarla posteriormente con un editor de texto.

Por ejemplo, para enviar la información que normalmente aparecería en la pantalla al fichero CHKOUT.TXT escriba la orden

```
A>CHKDSK /V >CHKOUT.TXT
A>
```

Una vez que la información está en el fichero CHKOUT.TXT puede imprimirla en la impresora o verla con la orden TYPE o con un procesador de texto.

Si el fichero al que usted está redirigiendo la información ya existe, el signo le dice al MS-DOS que lo borre y lo reemplace con los nuevos datos. Sin embargo, puede usar dos signos de mayor que (> >) si quiere añadir texto al final de un fichero existente.

Por ejemplo, para hacer un fichero llamado BC-DIR que tenga la información que la orden DIR escribe para B: y C:, dé las órdenes

```
A>DIR B: >BC-DIR
A>DIR C: >>BC-DIR
```

La primera orden comienza un nuevo fichero y la segunda añade texto al final de este fichero.

Un inconveniente al usar salida redirigida es que no aparece ningún texto en la pantalla hasta que el programa ha terminado. Por esta razón, usted no sabrá si el programa le ha hecho una pregunta y está esperando a que introduzca algo. De este modo, la salida redirigida sólo es útil para programas que no hacen preguntas mientras se ejecutan.

TUBOS (PIPES)

Una tercera característica avanzada del MS-DOS son los tubos. Un *tubo* (pipe) es una forma de conectar dos programas de forma que la salida de un programa sea la entrada del siguiente. Toda la información que el primer programa habría escrito en la pantalla se envía, en cambio, como entrada del

segundo programa, reemplazando la entrada por teclado. Es similar a redirigir la salida del primer programa hacia un fichero, y a continuación utilizar ese fichero como entrada redirigida para el segundo programa.

El método que se utiliza para conectar dos programas es muy simple: se pone una barra vertical (|) entre los dos programas en una única línea de orden. Se puede usar cualquier número de argumentos en cada orden, pero las dos órdenes que se van a conectar tienen que estar separadas por una barra vertical.

Desafortunadamente, este concepto tiene un uso muy limitado con el MS-DOS, puesto que muy pocos programas producen la entrada exacta que se necesita para ejecutar otro programa. Generalmente, sólo programas de utilidades especiales (como las órdenes SORT y FIND) son útiles en los tubos. El resto de esta sección da ejemplos de conexión de órdenes.

Puede conectar la orden FIND con la orden DIR para obtener información útil acerca de los ficheros del disco. Para encontrar los ficheros que fueron actualizados por última vez en una fecha particular (11-27-83, en este ejemplo), puede conectar la salida de la orden DIR a la orden FIND:

```
A>DIR | FIND "11-27-83"
HARDQUES DOC      1076   11-27-83    9:33p
NEWHARD  LET      1384   11-27-83    9:31p
```

También puede usar este método para encontrar los nombres de los subdirectorios puesto que sabe que el signo menor que sólo aparecerá en las líneas con nombres de subdirectorios:

```
A>DIR | FIND "<"

.              <DIR>      12-06-83     6:06p
..             <DIR>      12-06-83     6:06p
STEIN          <DIR>      12-06-83     6:07p
SMITH          <DIR>      12-06-83     6:09p
```

Esto lleva a una nueva forma de obtener un directorio de sus ficheros que ignora los subdirectorios:

```
A>DIR | FIND /V "<"
 Volume in drive A has no label
 Directory of  A:\text\clients

MAILLIST BAS        689    9-14-83    9:08a
HARDQUES DOC       1076   11-27-83    9:33p
NEWHARD  LET       1384   11-27-83    9:31p
MAILLIST DAT         89   12-05-83    9:48p
SALES    84          28   11-09-83   12:22p
%PIPE1   $$$          0    2-27-84    3:45p
%PIPE2   $$$          0    2-27-84    3:45p
        11 File(s)    317440 bytes free
```

Recuerde, de la descripción de la orden FIND en el capítulo 3, que el argumento /V de la orden FIND hace que la orden escriba las líneas que no contienen la cadena (en este caso, « < »).

Observe los dos ficheros situados al final del listado del directorio en el ejemplo. Los crea el MS-DOS cuando se usan tubos. Son sólo temporales y se borran cuando se termina la última orden de una lista de tubos. Desafortunadamente aparecen en los listados de directorios que usted realiza cuando está en un tubo.

La orden SORT puede usarse también con la orden DIR. Recuerde, sin embargo, que la orden SORT ordena por el código ASCII, lo cual a menudo es poco práctico.

Por ejemplo, para ordenar los ficheros de un directorio por su extensión, utilice la orden SORT con el argumento /+9. (Recuerde del capítulo 3 que la orden SORT clasifica las líneas empezando por la columna de la izquierda, a no ser que se le dé un número de columna donde comenzar).

```
A>DIR | SORT /+9

             <DIR>        12-06-83    6:06p
..           <DIR>        12-06-83    6:06p
STEIN        <DIR>        12-06-83    6:07p
SMITH        <DIR>        12-06-83    6:09p
%PIPE1    $$$          0   2-27-84    3:45p
%PIPE2    $$$          0   2-27-84    3:45p
SALES     84          28  11-09-83   12:22p
MAILLIST  BAS        689   9-14-83    9:08a
MAILLIST  DAT         89  12-05-83    9:48p
HARDQUES  DOC       1076  11-27-83    9:33p
NEWHARD   LET       1384  11-27-83    9:31p
       11 File(s)    317440 bytes free
 Volume in drive A has no label
 Directory of  A:\text\clients
```

Puesto que el carácter blanco precede a los caracteres alfabéticos y numéricos, las entradas de los subdirectorios se listan primero.

Se pueden enlazar muchas órdenes juntas con tal de que la salida de una sea siempre utilizable como entrada de otra.

Por ejemplo, se puede utilizar una única línea para listar ordenadamente todos los ficheros (no subdirectorios) de un directorio, sin incluir los ficheros temporales que se crean al usar tubos:

```
A>DIR | FIND /V "<" | FIND /V "e" | FIND "-" | FIND /V "%PIPE" | SORT
HARDQUES DOC     1076  11-27-83   9:33p
MAILLIST BAS      689   9-14-83   9:08a
MAILLIST DAT       89  12-05-83   9:48p
NEWHARD  LET     1384  11-27-83   9:31p
SALES    84        28  11-09-83  12:22p
```

En este ejemplo, las sucesivas órdenes FIND se usan para suprimir las líneas que no se quieren visualizar. Brevemente, FIND /V «<» elimina las entradas de subdirectorios; FIND /V «e» elimina las líneas que contienen el rótulo y el número de ficheros (puesto que ambas líneas contienen al menos una «e» minúscula); FIND «-» elimina las líneas en blanco (puesto que no contienen guiones como las líneas de nombres de ficheros); y FIND /V «%PIPE» elimina los ficheros temporales debidos a los tubos. A continuación, la orden SORT final ordena las líneas por el nombre y el tipo del fichero. Recuerde que la salida de cada orden FIND es el listado del directorio menos las líneas eliminadas por dicha orden FIND.

EDICION DE LAS LINEAS DE ORDENES DEL MS-DOS

Las órdenes avanzadas tratadas hasta ahora en este capítulo son todas nuevas en la versión 2 del MS-DOS. La última característica que vamos a tratar en este tema es una herramienta práctica, presente también en la versión 1. Esta característica es el uso de teclas especiales para editar líneas de órdenes.

Cada vez que se escribe una orden, el MS-DOS la guarda en una *plantilla,* que es un área especial de memoria que el MS-DOS utiliza para la edición de órdenes. Si quiere repetir la última orden que dio al MS-DOS, o si quiere dar una orden que es similar a la última orden, puede decirle al MS-DOS que haga una copia de esa orden y le deje editarla con teclas especiales. Desgraciadamente, estas teclas no son siempre útiles, puesto que el MS-DOS a veces «olvida» el significado de las teclas.

La tabla 4-1 muestra las teclas que se pueden usar para editar esta plantilla y el significado de cada tecla. Estas teclas coinciden con las del IBM PC y similares. Puesto que todos los teclados no son iguales, los nombres de las teclas de su computadora pueden ser distintos de los que se dan aquí.

La tecla que usará más a menudo es la tecla F3. Esta tecla se utiliza para copiar la línea de la orden, de la plantilla.

Tabla 4-1. Teclas de edición de órdenes

Tecla	Significado
F1	Copia el siguiente carácter de la plantilla (igual que la tecla →).
F2*x*	Copia de la plantilla todos los caracteres hasta el carácter *x* (no incluido).
F3	Copia de la plantilla los caracteres restantes.
F4*x*	Ignora todos los caracteres de la plantilla hasta el carácter *x* (no incluido).
F5	Copia la línea actual en la plantilla y comienza a editar de nuevo.
DEL	Ignora el siguiente carácter de la plantilla.
INS	Comienza el modo de inserción; todos los caracteres introducidos, hasta que se pulse la siguiente tecla de edición, son insertados.
BACKSPACE	Retrocede un carácter (igual que la tecla ←).
ESCAPE	Descarta la línea actual y comienza a editar de nuevo.

Por ejemplo, suponga que ha dado la orden equivocada

```
A>DIT
Bad command or file name
```

Usted pretendía que el nombre del programa fuera DIR y no DIT. El MS-DOS intenta encontrar un programa llamado DIT y no lo consigue. Ahora, en lugar de tener que escribir la línea de la orden de nuevo, puede pulsar simplemente la tecla F3:

```
A>DIT
```

El MS-DOS copia la plantilla (la última orden) en la línea de la orden y deja el cursor al final de la línea. A continuación, pulse una vez la tecla BACKSPACE, con lo cual se borra la T:

```
A>DI
```

Ahora, simplemente escriba la R para producir la orden que quería.

Las otras teclas de edición son también fáciles de usar. Por ejemplo, si la última orden que dio usted fue TYPE C:ACCNTS.RPT y en cambio quería

escribir el fichero de B:, debería pulsar la tecla F1 cinco veces, escribir B y, a continuación, pulsar la tecla F3. Esto produciría la línea requerida. Debería experimentar con estas teclas durante un tiempo hasta que se sienta cómodo con ellas.

Un problema con las teclas de edición es que, a veces, el MS-DOS las «olvida». La razón de que esto ocurra es demasiado técnica para explicarla largamente aquí, pero se dará cuenta inmediatamente de si las ha olvidado, puesto que al pulsarlas no se obtiene ningún resultado.

5

TRATAMIENTO DE INCIDENTES

Algunos problemas típicos
Cuidado de los discos
Cómo evitar problemas graves haciendo copias de seguridad
Qué hacer con los mensajes de error
Recuperación después de problemas graves

Todo sistema operativo puede fallar en diferentes ocasiones por diferentes motivos. Algunos sistemas operativos, como el MS-DOS proporcionan unas cuantas posibilidades de solución para los diferentes tipos de riesgos en el uso de la computadora. Otros, tales como el CP/M, virtualmente no dan al usuario ninguna ayuda para solucionar un incidente. Tanto si los problemas han sido originados por errores humanos, como por fallo del hardware, errores del software o cortes de energía, pueden causar la pérdida de muchas semanas de trabajo. Sin embargo, tales pérdidas se pueden prevenir teniendo un poco más de cautela. Este capítulo le avisa de los riesgos potenciales y le da indicaciones de cómo evitarlos.

ALGUNOS PROBLEMAS TIPICOS

Cuanto más se usan las computadoras, más se depende de su fiabilidad. Desgraciadamente, la fiabilidad de una computadora se basa en que centenares de factores funcionen correctamente. El mal funcionamiento de alguno de

estos factores puede causar serios problemas. La gravedad del problema resultante es una combinación de la cantidad de información o trabajo que se ha perdido y, naturalmente, de la cantidad que puede ser recuperada.

Algunos de los errores que probablemente va a encontrar son:

- *Disco en mal estado.* Los errores de los discos son los problemas más frustrantes en casi todas las computadoras, puesto que no se pueden evitar y pueden causar la pérdida de grandes cantidades de datos. Aún teniendo una experiencia excelente en el manejo de discos, un pequeño error en el disco puede hacer que todo el disco sea ilegible.

- *Fichero borrado accidentalmente.* Desgraciadamente no hay una orden de «desborrar» en el MS-DOS. Algunos paquetes de utilidad permiten recuperar ficheros borrados, en ciertas circunstancias (usualmente, si no se ha vuelto a escribir en el disco desde que el fichero fue borrado). La orden ERASE no es la única orden peligrosa en el MS-DOS; se puede destruir accidentalmente un fichero con la orden COPY, si se copia en un disco que ya tiene un fichero con el mismo nombre. Naturalmente, formatear un disco que contiene información importante puede también ser desastroso.

- *Programa congelado.* Su computadora puede parecer que está completamente parada mientras ejecuta un programa de aplicación (como por ejemplo, un procesador de texto o un programa de contabilidad): la pantalla no cambia, los discos no se mueven (o simplemente siguen girando) y al pulsar teclas no cambia nada. Estos problemas casi siempre se deben a un error en el programa o a que el programa no tenía previstas todas sus posibles respuestas. Normalmente, en el peor caso, lo que se pierde es el trabajo invertido en la ejecución del programa.

- *Fallo del hardware.* Aunque la mayor parte del hardware de las computadoras es muy fiable, puede a veces dejar de funcionar. Cuando esto sucede, suele ser muy difícil determinar exactamente qué parte no funciona. La mayoría de las computadoras tienen sólo unas cuantas partes móviles (como por ejemplo, las unidades de disco), y encontrar las averías en ellas es normalmente más fácil que determinar si un chip se ha fundido o encontrar un cortocircuito en una tarjeta de circuito impreso.

- *Corte de energía.* Un corte de energía o incluso una ligera bajada de potencia, mientras se está ejecutando un programa, puede hacer que se pierda el trabajo que se acaba de realizar. También puede dañar los ficheros si se estaba escribiendo en el disco en ese momento o si la cabeza del disco estaba colocada sobre una parte importante del disco.

Naturalmente, hay otra multitud de errores que pueden causarle serios apuros. Este capítulo describe, tanto las medidas preventivas normales que se

deberían tomar para evitar el peor desastre (la pérdida de todos los datos) como los problemas corrientes que pueden surgir y cómo salir de ellos.

CUIDADO DE LOS DISCOS

Una de las mejores formas de evitar desastres es cuidar los discos. Es mucho mejor prevenir los daños que tener que solucionarlos. He aquí algunos consejos:

- Nunca toque las partes expuestas de un disquete flexible. Debido a que los dedos están siempre recubiertos de una fina capa de aceites naturales (sin mencionar el polvo y las partículas ambientales), incluso un simple roce puede hacer inaccesibles para el MS-DOS una gran cantidad de datos. Cuando saque el disquete de la unidad de disco vuelva a colocarlo siempre en su funda de papel.

- No ponga los disquetes encima de la computadora porque podrían ser dañados por las ondas magnéticas, ocasionalmente intensas, que hay a menudo alrededor de las fuentes de alimentación. Puesto que la información de los discos se almacena magnéticamente, una fuente magnética fuerte podría borrar los datos del disquete. Asegúrese de que ha eliminado todos los imanes aparentemente inofensivos (como los que sujetan el papel en un caballete para máquina de escribir).

- No deje los discos al sol o cerca de fuentes de calor como radiadores, calentadores de agua y lámparas. Los plásticos del disco son sensibles al calor y pueden deformarse.

- No escriba en la etiqueta exterior de un disquete con algo distinto de un rotulador de punta de fieltro. Los lápices y los bolígrafos pueden abollar la superficie magnética y esto provocaría errores cuando el MS-DOS lea o escriba datos.

- Cuando mueva una computadora que tenga un disco rígido, siempre levántela. Nunca la deslice, puesto que la base de goma causaría fuertes vibraciones cuando se desliza la computadora sobre una superficie. Esta vibración puede dañar la cabeza y el material magnético.

Naturalmente, hay muchas más precauciones que se pueden tomar para evitar que se dañen los discos. Debería leer el manual del hardware de su computadora, prestando especial atención a las secciones de advertencias y precauciones.

COMO EVITAR PROBLEMAS GRAVES HACIENDO COPIAS DE SEGURIDAD

Otra forma importante de evitar los problemas graves es hacer regularmente copias de seguridad de los discos. Una computadorea puede «*caerse*» (tener un fallo de funcionamiento) a pesar de las mejores precauciones. Sin embargo, si se hacen regularmente copias de los discos de programas y de datos, el daño será menor.

Casi todos los profesionales de la informática recalcan la importancia de copiar regularmente los datos en otro disco, denominado *disco de seguridad.* El disco debería guardarse en un lugar seguro. La seguridad es, naturalmente, relativa: si su oficina se inunda o se quema y el disco de seguridad está en ella, obviamente no le será de mucha ayuda. Sin embargo, para la mayoría de los casos, simplemente el hacer una copia de seguridad del fichero suele ser una precaución adecuada. El proceso de hacer copias de seguridad de ficheros se llama también a veces *archivar.*

Usted tiene que decidir qué ficheros necesita copiar, con qué frecuencia debe hacer copias de seguridad y qué método va a utilizar. Las dos primeras decisiones se pueden tomar muy fácilmente: copiar todos los ficheros de la computadora, al menos una vez y copiar cada fichero que se modifique tan a menudo como usted tenga tiempo para hacerlo. La última decisión, qué método utilizar, depende del hardware de su computadora.

Elección de un método de realización de copias de seguridad

El método de copia que usted use dependerá del hardware de su sistema. La mayoría de las computadoras MS-DOS tienen sólo unidades de disco rígido y flexible; otras también tienen unidades de cinta. Un resumen de los métodos sugeridos en esta sección se da en la tabla 5-1.

Tabla 5-1. Métodos para hacer copias de seguridad de los datos

Origen de la copia	Destino de la copia	Método sugerido
Disquete	Disquete	Ordenes DISKCOPY o COPY.
Disco rígido	Disquete	Orden COPY o utilidad distribuidora por el vendedor de la computadora.
Disco rígido	Cinta	Utilidad distribuidora por el vendedor de la unidad de cinta.

Si su sistema tiene sólo disquetes flexibles, la orden DISKCOPY es una forma rápida y fácil de copiar disquetes enteros. La orden DISKCOPY puede hacer una copia completa de cada uno de sus disquetes y es muy fácil de usar. Vea el capítulo 3 para una descripción completa de la orden.

Después de haber hecho varias copias de seguridad de un disquete particular, tendrá muchos disquetes con copias de seguridad. Puesto que el conservar muchas copias del mismo fichero en varios disquetes es costoso y puede causar confusión, tiene que decidir cuánto tiempo va a conservar cada copia de seguridad.

Un buen método para controlar las copias de seguridad es tener dos conjuntos incrementables de disquetes de seguridad indicados como *X* e *Y*. Cuando realice la copia de seguridad debería alternar entre estos dos conjuntos de disquetes. Por ejemplo, si la última copia de seguridad la hizo en los disquetes *X*, entonces la próxima copia la hará en los disquetes *Y*. Esto asegura que, si tanto el disquete maestro como el disquete de seguridad se estropean por alguna razón, todavía queda una versión ligeramente más antigua del disquete.

Si sólo cambia unos cuantos ficheros del disquete, no es necesario copiar todo el disquete con la orden DISKCOPY. En vez de esto, puede utilizar la orden COPY para copiar aquellos ficheros que han sido modificados. Una forma rápida de comprobar qué ficheros hay que copiar, es mirar las fechas y horas de las últimas actualizaciones con la orden DIR. Para estar seguro de que la copia con la que se está trabajando es siempre la más reciente, debería tomar nota de cuando hizo la última copia de seguridad de cada fichero.

Existen otras órdenes además de COPY y DISKCOPY para hacer copias de seguridad de discos rígidos. Muchos discos rígidos vienen con un programa para copiar ficheros del disco duro a los disquetes. Por ejemplo, IBM suministra dos programas, BACKUP y RESTORE. Sin embargo, los usuarios de IBM deberían saber que estos programas no son fáciles de usar. Se puede perder información si no se utilizan los disquetes exactos que dice IBM. Si no tiene un programa especial para copiar del disco duro, use la orden COPY para copiar cualquier fichero que haya modificado después de la última copia de seguridad.

Algunos sistemas de disco rígido, tales como los de Tecmar y Xcomp tienen discos que se pueden sacar (llamados *discos rígidos extraíbles*). Si su sistema tiene dos de estos discos, puede copiar un disco en otro, utilizando DISKCOPY o una orden equivalente suministrada con las unidades de disco. Aunque haga regularmente copias de seguridad de los ficheros de discos rígidos, no se olvide de hacer también copias de los ficheros que haya creado en disquetes.

Las *unidades de cinta magnética* son otro método de hacer copias de seguridad de los discos rígidos. Estas unidades de cinta utilizan grandes cassettes o bobinas y pueden almacenar datos con mucha más exactitud que las pequeñas cassettes. Son extremadamente rápidas y permiten copiar una gran cantidad de datos en ellas. Por ejemplo, muchas unidades de cinta son capaces de copiar un disco rígido de 10 megabytes entero en menos de 5 minutos. Si tiene

una unidad de cinta magnética en su computadora, probablemente vendrá con un programa que haga copias de seguridad de una variedad de tipos de discos.

Se debe hacer una copia de seguridad de cada fichero de texto o de datos, siempre que sea posible. Por ejemplo, si usa constantemente una serie de ficheros de datos para su sistema de contabilidad, debería copiar los ficheros dos veces al día. De esta manera, perderá sólo medio día de trabajo si se pierde o se estropea un fichero. Si sólo actualiza los ficheros una vez a la semana, perderá una semana de trabajo. Naturalmente, si nunca hace copias de seguridad de los datos, posiblemente perderá su negocio.

Algunos ficheros nunca cambian. Por ejemplo, un fichero puede ser una carta que escribió y que ya ha enviado. No hay razón para copiar tales ficheros más de una vez. Sólo necesita copiar ficheros que hayan cambiado desde la última vez que hizo una copia de seguridad.

Copias de seguridad del software de aplicación

La mayoría del software que se compra o que viene con la computadora se distribuye en disquetes. Puesto que los disquetes se desgastan, o bien pueden dañarse, debe hacer siempre una copia completa de cada disquete de software original (el *disquete maestro*) antes de usar el software. A continuación, guarde el maestro en un sitio seguro y utilice en su lugar la copia. Si su copia de trabajo se vuelve ilegible a causa del deterioro o de la mala utilización, puede simplemente copiar el maestro de nuevo. Debería poner una etiqueta de protección de escritura en el disco maestro a no ser que las instrucciones del software digan lo contrario.

Los fabricantes de software suelen incluir instrucciones para hacer copias de seguridad del disquete maestro. Si no trae instrucciones, pruebe con las órdenes COPY o DISKCOPY descritas en el capítulo 3. Es probable que no pueda copiar el disco: las órdenes informarán de errores tales como «Sector not found» (Sector no encontrado). La mayoría de los fabricantes de software utilizan la protección contra copia para evitar que se pueda copiar el disquete. Lo hacen para limitar el número de copias ilegales que se hacen de su software. Muchos de estos fabricantes incluyen copias de seguridad del software cuando se compra el paquete, y otros envían copias de seguridad cobrando una cierta tarifa.

¿Qué se puede hacer si el disquete está protegido contra copia y no se puede conseguir una copia de seguridad? Hay que estar seguro de que el disquete original no falle nunca, o ya no se podrá usar el software. Sin embargo, puesto que todos los disquetes fallan alguna vez, esta situación es difícilmente deseable. La siguiente sección muestra un método que puede usar si no puede conseguir una copia de seguridad adecuada de su software de aplicación.

Copia de los disquetes protegidos contra copia

Existen en el mercado un cierto número de programas que permiten hacer copias de disquetes protegidos contra copia. Si posee algún software protegido contra copia, un programa de copia de discos le puede proporcionar un seguro contra el desastre.

Naturalmente, no debe usar uno de estos programas para hacer copias de software que de otra manera se hubieran vendido. Tal acción es ilegal, de la misma forma que es ilegal copiar un libro en una fotocopiadora en vez de comprarlo en la tienda. En realidad, puesto que copiar software es más fácil y más barato que copiar un libro, la copia ilegal es mucho más perjudicial para los vendedores de software que para los editores de libros.

Un programa popular es *Copy II PC* de Central Point Software. Hace copias de seguridad de más de 100 programas protegidos y es muy fácil de usar. Actuando como un sustituto de la orden DISKCOPY, *Copy II PC* incluso verifica automáticamente el disco copiado. También incluye una utilidad para comprobar la velocidad de las unidades de disco. Otros programas similares son *SafeCopy* de Computer Shack, Inc., *PC Duplicator* de Logical Systems y *Savior* de Omega MicroWare, Inc. Algunos de los programas para la conservación de discos, mencionados más adelante en este capítulo, también copian discos protegidos.

Debería investigar un poco antes de comprar un programa de copia de discos, para estar seguro de que el programa puede copiar el producto software, del que usted necesita hacer una copia de seguridad, y de que se puede utilizar en su computadora. La mayoría de estos productos sólo funcionan en el IBM PC.

QUE HACER CON LOS MENSAJES DE ERROR

Las copias de seguridad de los discos proporcionan un «seguro» contra la pérdida de demasiados datos importantes. Otra forma de seguro es la capacidad de reconocer mensajes de error, porque si se entiende un problema mientras está ocurriendo, se puede corregir antes de que sea demasiado tarde.

Los mensajes de error pueden aparecer por una gran variedad de razones que implican tanto al hardware como al software. Las causas de algunos fallos del hardware son obvias: café derramado sobre un disquete o en la computadora, o una unidad de disco estropeada. La mayoría de las veces, sin embargo, no es obvio por qué ha ocurrido un error. Muchos programas de aplicación dan misteriosos mensajes de error, si es que los dan; los mensajes de error del MS-DOS suelen ser más fáciles de reconocer.

Si entiende los mensajes de error que le salgan, podrá prevenir las equivocaciones repetitivas. Esto es debido a que los mensajes de error ayudan frecuentemente a entender la forma correcta de hacer lo que se quería. Si el error

fue causado por algo fuera de su control, los mensajes suelen decirle cómo salir del problema.

Cómo entender los mensajes de error del MS-DOS

Los mensajes de error del MS-DOS suelen ser mejores que los que dan otros sistemas operativos, pero aún así pueden confundirle si no está familiarizado con ellos. A todo el mundo le salen mensajes de error a causa de las equivocaciones en la escritura de órdenes o argumentos. No se preocupe si le sale uno.

La tabla 5-2 lista los mensajes de error más comunes del MS-DOS, sus posibles causas y sus posibles soluciones. Esta lista no es exhaustiva; hay docenas de mensajes de error que le pueden salir. Algunas órdenes (especialmente CHKDSK) tienen numerosos mensajes de error que, aunque importantes, probablemente son ininteligibles para cualquiera, excepto para programadores avanzados. Sin embargo, esta lista le puede ayudar a reconocer alguno de los errores que son frecuentes cuando se utiliza el MS-DOS. Muchos de los mensajes de error de disco que aparecen en la tabla le hacen la pregunta

```
Abort, Retry, Ignore?
```

Esto significa que el MS-DOS le está dando otra posibilidad para leer o escribir en el disco. Usted escribe la primera letra de una de las tres elecciones posibles. «Abort» indica que usted no quiere intentar leer o escribir en el disco de nuevo. «Retry» significa que cree que el MS-DOS podría tener éxito si lo intentara de nuevo (por ejemplo, si usted había dejado abierta la puerta de la unidad). «Ignore» le dice al MS-DOS que olvide lo que estaba intentando; esto casi siempre causa el mismo resultado que «Abort», puesto que es raro que un programa pueda seguir correctamente sin haber leído o escrito la información que pretendía.

Con algún esfuerzo, casi siempre es posible entender los mensajes de error del MS-DOS y determinar qué es lo que hay que hacer para poder seguir. Cuando se encuentre con estos errores debe recordar lo que los causó e intentar prevenirlos en el futuro. Se pueden producir mensajes de error de disco, tanto desde el MS-DOS como desde programas de aplicación.

Mensajes de error de los programas de aplicación

Muchos mensajes de error de los programas de aplicación son más difíciles de entender que los del MS-DOS. Si no los entiende, también tendrá dificultades para salir de los problemas que describen. Cuando busque respuestas en su manual, quizá encuentre que raramente se explican las causas del error.

Algunos mensajes de error son más oscuros que otros, conteniendo a menudo números y letras ininteligibles (como por ejemplo, «Error 43: interrupt exception at 4AC2» que significa «excepción de interrupción en 4AC2»).

Tabla 5-2. Mensajes de error más comunes y sus soluciones

Mensajes de error	Causa probable	Posible solución
«Bad command or file name» (Error en la orden o en el nombre de fichero)	El MS-DOS no pudo encontrar la orden dada por usted.	Compruebe si escribió correctamente el nombre de la orden. Si es externa, asegúrese de que el fichero de la orden está en el disco por defecto.
«Bad floppy diskette» (Error en el disquete)	El MS-DOS está intentando formatear un disquete que tiene una etiqueta de protección de escritura, o bien el disquete no está rotando correctamente en su envoltura.	Compruebe si tiene la etiqueta de protección de escritura. Si no la tiene tire el disquete para que no le cause problemas posteriormente.
«Disk error reading (o writing) drive *x*» (Error de disco al leer o escribir en la unidad *x*)	El MS-DOS no puede leer información de la unidad.	Puede ser causado por un gran número de problemas diferentes. Puede haber un sector en mal estado o el directorio del disco no está bien. Pruebe a ejecutar el programa CHKDSK con el disco.
«Duplicate file name or File not found» (Nombre de fichero duplicado o fichero no encontrado)	Usted ha intentado cambiar de nombre un fichero y darle un nombre de fichero que ya existía en el disco, o bien ha intentado cambiar de nombre un fichero que no está en el disco.	Elija un nombre diferente o compruebe si introdujo el nombre que pretendía.
«Error reading FAT 1 (o FAT 2)» (Error al leer FAT 1 ó FAT 2)	La tabla de asignación de ficheros de su disco está estropeada.	Es un error serio. Utilice la orden COPY para copiar del disco tantos ficheros como pueda y a continuación tírelo (si es un disquete) o vuelva a formatearlo (si es un disco duro).

Tabla 5-2. Mensajes de error más comunes y sus soluciones *(continuación)*

Mensajes de error	Causa probable	Posible solución
«File allocation table bad, drive *x*» (Tabla de asignación de ficheros de la unidad *x* en mal estado)	La tabla de asignación de ficheros de su disco está estropeada.	Es lo mismo que «Error reading FAT 1 (o FAT 2)».
«File cannot be copied to itself» (No se puede copiar un fichero sobre sí mismo)	COPY no puede copiar un fichero encima de él mismo.	Dé a COPY un nombre de destino diferente del nombre origen, o use el mismo nombre, pero en disco diferente.
«File not found» (Fichero no encontrado)	Uno de los ficheros mencionados en una orden no pudo ser encontrado.	Compruebe que el fichero existe, que escribió su nombre correctamente y que dio los argumentos en el orden correcto.
«Invalid directory» (Directorio no válido)	Ha nombrado un subdirectorio que no existe.	Compruebe que existe el camino y que ha escrito correctamente cada nombre del camino.
«Not ready error reading (o writing) drive *x*» (Error, la unidad *x* no está preparada para la lectura o escritura)	La puerta de la unidad no está cerrada o no hay disquete en la unidad.	Compruebe la unidad y asegúrese de que el disquete está orientado correctamente.
«Write protect error writing drive *x*» (Error de protección de escritura al escribir la unidad *x*)	El MS-DOS no puede escribir en un disco que tiene protección de escritura.	Compruebe primeramente si realmente quiere escribir en ese disco. En ese caso, quite la etiqueta de protección de escritura.

La mayor parte del software viene con documentación; una buena documentación debería listar todos los mensajes de error que se pueden encontrar. Desgraciadamente, muchos programas populares tienen manuales con escasas o inexistentes definiciones de los mensajes de error, de forma que usted tendrá que intentar descifrar por sí mismo lo que significa un mensaje de error. Si está en esta situación, he aquí algunas indicaciones:

- *Lea el mensaje de error cuidadosamente antes de que le entre el pánico.* Por ejemplo, el mensaje «FATAL ERROR 12: Cannot write file on B:»

(Error fatal 12: No se puede escribir el fichero en B:) puede parecer terrible a simple vista, pero puede ser simplemente la forma que tiene el autor del software de decirle que cierre la puerta de la unidad B:. Si un mensaje de error le dice que no puede hacer algo en una unidad de disco, asegúrese de que la unidad está preparada y de que el disco correcto está en la unidad. Si el programa está intentando escribir un fichero en el disco, asegúrese de que el disco no tiene protección de escritura y de que no está lleno.

- *Póngalo por escrito.* Siempre que vea un mensaje de error en la pantalla que no le es familiar, anótelo exactamente como aparezca. Si el error resulta ser importante, tendrá que conseguir más ayuda del vendedor de la computadora, el cual necesitará el mensaje de error completo para descifrar cuál fue el problema.
- *Lea su manual.* Puede que haya un capítulo o apéndice, en su manual de software, sobre los mensajes de error; si no, puede que encuentre la información que necesita en algún otro capítulo relevante. Por ejemplo, si su procesador de texto le da un mensaje como «Can't find WRITE.OVR» (No puedo encontrar WRITE.OVR), lea el capítulo de instalación para saber dónde está buscando el fichero. Releyendo rápidamente los primeros dos capítulos y los apéndices de un manual se clarifican la mayor parte de los errores comunes.
- *Pregunte a otra persona que utilice su computadora o su software.* Otras personas familiarizadas con el programa pueden haber tenido el mismo problema. Si no, al menos, quizá puedan adivinar su causa.

  Puesto que el error puede significar que usted pierda horas de trabajo, su enfado y su frustración pueden hacer que ignore una solución razonable. Haga una pausa y vuelva a ello posteriormente; una mente despejada siempre ayuda en problemas que, a menudo, pueden ser resueltos de forma lógica.
- *Llame al distribuidor que le vendió el software.* Si el error ocurrió cuando dio una orden del MS-DOS y no está explicado en el manual de su computadora o en este libro, llame al distribuidor de su computadora (vea más adelante la sección de este capítulo «Dónde encontrar ayuda»). El o ella probablemente estará familiarizado con los errores más comunes.

RECUPERACION DESPUES DE PROBLEMAS GRAVES

Ahora que ha aprendido algunas formas de evitar desastres y de diagnosticar problemas, es menos probable que vaya a ser víctima de desastres informáticos. Sin embargo, no siempre se pueden evitar los problemas y también

debería saber cómo salir de aquéllos que aparecen durante el uso normal de la computadora. Esta sección explica los métodos que se pueden usar para restaurar datos que se han vuelto ilegibles y la mejor forma de conseguir ayuda en otros problemas.

Recuperación de discos en mal estado

La mayor parte de los sistemas operativos no proporcionan una forma de recuperar los datos de un disco dañado o ilegible. Es una pena, porque el que un disco se dañe es un problema frecuente. El MS-DOS proporciona unas cuantas órdenes sencillas para ayudarle a recuperar un disco en mal estado; sin embargo, algunos problemas del disco requieren soluciones más técnicas. En estos casos más serios, algunas de las utilidades mencionadas en la siguiente sección pueden ayudarle. Desgraciadamente, para utilizar dichas utilidades hay que tener un mayor conocimiento técnico que el que requieren las órdenes del MS-DOS.

Si el disco que quiere utilizar tiene áreas ilegibles, la primera decisión que necesita tomar es si merece la pena conservarlo. Por ejemplo, si hay sólo un fichero en el disco y usted tiene todavía una copia del fichero, probablemente es más fácil tirar el disco que intentar recuperarlo.

Lo más frecuente es que el disco tenga datos importantes. Si este es el caso, el MS-DOS tiene dos órdenes para recuperar discos en mal estado. Estas órdenes son CHKDSK, utilizada con el argumento /F, y RECOVER. La diferencia entre ellas tiene que ver con los ficheros que intentan recuperar: CHKDSK se utiliza para recuperar ficheros cuyas entradas en el directorio contienen información incorrecta y RECOVER para recuperar ficheros que están en sectores dañados.

Puede que le salga un mensaje de que el MS-DOS no puede leer parte del disco (como por ejemplo, «Disk error reading B:», que significa «Error de disco al leer B:»). Para salir de este problema, primero intente utilizar la orden CHKDSK. Si esta orden indica que hay errores en el disco, dé la orden de nuevo con el argumento /F. Si los mensajes de la pantalla no dicen que el disco está arreglado, pruebe con la orden RECOVER.

Algunos errores son más serios, haciendo inutilizable el disco entero. Si es este el caso, las órdenes CHKDSK y RECOVER no le serán de utilidad. Por ejemplo, el mensaje «Disk error reading FAT 1» seguido por «Disk error reading FAT 2» indican que la tabla de asignación de sus ficheros está tan estropeada que el MS-DOS no puede descifrar lo que debe hacer. La *tabla de asignación de ficheros* es la parte del directorio que le dice al MS-DOS dónde reside dentro del disco cada parte de cada fichero. Si una parte de la tabla está en mal estado, todos los ficheros de esa parte son ilegibles.

Sin embargo, en la mayoría de los casos de tablas de asignación de ficheros en mal estado, aún se pueden leer la mayor parte de los ficheros del disco, pero probablemente no se puede escribir en éste. Después de transferir tantos

ficheros como pueda con la orden COPY, debería tirar el disquete. Si el error FAT ocurre en el disco duro, vuelva a formatearlo.

Aunque el MS-DOS le proporciona las órdenes CHKDSK /F y RECOVER, éstas son totalmente inadecuadas cuando ocurre un error serio. En la siguiente sección verá que hay algunos programas que pueden recuperar grandes cantidades de trabajo perdido.

Utilidades de conservación de disco

Cuando las órdenes proporcionadas por el MS-DOS no son suficientes, se pueden comprar programas de otros vendedores que ayudan a recuperar discos ilegibles. Cuando compre utilidades, debe tener en cuenta que algunos programas son poco más que «desborradores» de ficheros, mientras que otros son series de programas que permiten ver los bytes del disco, analizar los ficheros dañados y otras cosas.

El principal inconveniente de estos paquetes es que asumen que usted tiene una gran cantidad de conocimientos técnicos para entender la información. Sin embargo, si su tiempo es importante para usted, debería comprar al menos uno de los paquetes y tenerlo a mano en caso de un fallo del disco. Puede que sea capaz de usar la utilidad con la documentación, o bien puede pedir ayuda exterior.

De los muchos paquetes a la venta, *Norton Utilities* de Peter Norton es el más completo y el más fácil de utilizar. La extensiva documentación está bien escrita y es muy informativa, y los programas permiten recuperar algunos ficheros borrados accidentalmente. También permiten realizar las modificaciones del disco, necesarias para la mayor parte de las recuperaciones, en los casos en los que ha sido dañado. Peter Norton es una de las personas más conocidas en el mundo del IBM PC, a causa de la gran utilidad de estos programas, así como por sus excelentes libros. *Disk Mechanic* de MLI Microsystems es un paquete similar dirigido a un mercado más técnico. El paquete incluye una utilidad para hacer copias de seguridad de discos protegidos contra copia.

Estos dos paquetes de utilidad sólo funcionan en el IBM PC. Si tiene otro sistema MS-DOS, pregunte a su distribuidor si alguno de los paquetes está disponible para su sistema. Las direcciones de los fabricantes de estos productos se dan en el apéndice E.

Dónde encontrar ayuda

Cuando todo lo demás falla, puede llamar a la tienda que le vendió la computadora o el software. Explicar el problema por teléfono puede ser muy difícil. Por esta razón, antes de llamar al distribuidor póngase en el lugar de éste e imagine qué información necesitaría para entender el problema. Anóte-

lo todo, aunque esté seguro de todos los hechos. Si no hay nada más, esto hará que el distribuidor esté más seguro de que usted tiene una bien meditada razón para llamarle.

Cuando necesite ayuda, primero determine con quién debería contactar. Si está ejecutando un programa de aplicación, llame al distribuidor que se lo vendió. Si es un error de hardware que no puede solucionar, llame al distribuidor del hardware. Si el error parece estar en el MS-DOS (lo cual es excepcionalmente raro), llame a su distribuidor de hardware si el sistema operativo venía con la computadora. Llamar al distribuidor suele ser más efectivo que contactar con los fabricantes del software o del hardware, puesto que, usualmente, la primera reacción de éstos consiste en decirle que llame al distribuidor.

La mejor solución es que haga todo lo que pueda usted mismo. Si tiene copias de seguridad recientes, aunque haya tenido un grave problema, no perderá mucho trabajo. La mayoría de la gente comienza a hacer copias de seguridad después del primer desastres; los que comienzan a guardar copias de seguridad antes del primer fallo del hardware o del software indudablemente perderán menos trabajo que los que esperan.

6

PROGRAMAS DE APLICACIONES PARA EL MS-DOS

El mercado actual de software para el MS-DOS
Cómo comprar software
Software de procesamiento de textos
Hojas de cálculo
Sistemas de gestión de bases de datos
Software integrado
Programas de contabilidad
Programas de planificación de proyectos
Programas gráficos
Software de comunicaciones
Aprendizaje asistido por computadora
Programas que mejoran el MS-DOS
Juegos y entretenimientos
Lenguajes de programación
Otros programas

Miles de programas están disponibles para el MS-DOS. Este capítulo trata de muchos de estos programas y describe la innumerable cantidad de cosas que se pueden hacer con el software ejecutable en una computadora MS-DOS:

- Entender los muchos tipos de software que están disponibles y saber qué tareas puede hacer la computadora con ellos.
- Saber los puntos buenos y malos de unos cuantos paquetes software de cada grupo especializado, de forma que se pueda comparar un paquete con otros que estén a la venta.
- Evitar contratiempos al comprar software, tales como precios exorbitante y escasa documentación.
- Elegir de entre los muchos paquetes generales a la venta aquéllos que resuelvan sus problemas de negocios.

Este capítulo sólo cubre la compra de software; el apéndice D proporciona la información necesaria para la compra de hardware.

El objetivo principal de este capítulo consiste en ayudarle como consumidor, comparando muchos programas. Los precios específicos no se mencionan, puesto que el precio de cada programa puede variar demasiado. Naturalmente, siempre debe comprobar los precios de programas competidores, a fin de que pueda comparar sus necesidades con el coste del software. Puede encontrar fácilmente las escalas de precios generales mirando los anuncios de las revistas de computadoras, como las que se mencionan en el capítulo 7.

La figura 6-1 lista los principales tipos de software que se tratan en este capítulo. Las secciones se presentan por orden de importancia para el lector medio. Puede que quiera hojear las discusiones del software que no está relacionado con usted (en vez de saltárselas), para el caso de que posteriormente se interese en los temas allí tratados.

Software de procesamiento de textos
Hojas de cálculo
Sistemas de gestión de bases de datos
Software entregado
Programas de contabilidad
Programas de planificación de proyectos
Programas gráficos
Software de comunicaciones
Aprendizaje asistido por computadora
Programas que mejoran el MS-DOS
Juegos y entretenimientos
Lenguajes de programación
Otros programas

Figura 6-1. Vista general del capítulo 6

Los nombres de los paquetes de software y sus fabricantes se mencionan a lo largo del capítulo. Si quiere una lista alfabética de estos nombres y direcciones (y los del capítulo 5) vea el apéndice E.

EL MERCADO ACTUAL DE SOFTWARE PARA EL MS-DOS

La mayor parte del software escrito para microcomputadoras, que no es para juegos, es para usuarios del MS-DOS. Las compañías que estuvieron limitadas por las características de otros sistemas operativos como CP/M, ahora están usando las ampliadas capacidades del MS-DOS para crear mejores programas. Nuevos paquetes se introducen cada día y muchos paquetes, que se ejecutan sobre otros sistemas operativos, están siendo convertidos al MS-DOS.

Desgraciadamente, el mercado del software para el MS-DOS está aún dominado por el IBM PC. Esto significa que más de la mitad de los programas disponibles para el MS-DOS sólo son ejecutables sobre el IBM PC o similares al PC (computadoras que internamente son similares al IBM PC). El resultado es que muchas computadoras MS-DOS no pueden ejecutar algunos de los mejores programas.

Con sólo unas cuantas pequeñas adiciones, la mayor parte del software MS-DOS diseñado para el IBM PC puede ejecutarse sobre la mayoría de las otras computadoras MS-DOS. Sin embargo, las compañías de software no hacen estas adiciones ni dan ayuda a los que no poseen un IBM. Así, si no posee un IBM (o similar), tiene siempre que comprobar la compatibilidad del software con su computadora.

COMO COMPRAR SOFTWARE

Cuando lea este capítulo, descubrirá muchos paquetes de software interesantes. Sin embargo, no es recomendable que usted simplemente llame al distribuidor de su computadora y pida un programa que parece interesante. Hay muchos factores que debería considerar antes de comprar un programa particular:

- *Coste en dólares.* Algunos paquetes de software cuestan menos de 100 $ y otros cuestan bien por encima de 1.000 $. La mayoría del software del MS-DOS, excepto el de juegos y muchos de los programas educativos, cuesta entre 250 $ y 600 $ por paquete (aunque algunos buenos programas son gratuitos).
- *Coste en tiempo.* Esto se refiere al tiempo gastado aprendiendo a usar el software. Si usted ya está familiarizado con el concepto general que hay

detrás de un paquete, podrá aprender a usarlo más rápidamente que si el tema le resulta nuevo. Puede aprender lo básico de algunos paquetes en una hora o dos; sin embargo, cinco a diez horas es una estimación más probable para el aprendizaje de paquetes sofisticados. Por ejemplo, los programas que siempre listan todas las órdenes posibles en la pantalla, llevan mucho menos tiempo de aprendizaje que los que no lo hacen.

- *Competencia.* Casi todos los paquetes de software a la venta tienen al menos un competidor (algunos tienen docenas). Debe siempre intentar encontrar unos cuantos paquetes de la competencia para compararlos con aquel en que está interesado.
- *Asistencia técnica proporcionada por el distribuidor.* Esto se refiere a la cantidad de responsabilidad que el distribuidor del programa quiere aceptar. La asistencia es importante porque usted probablemente querrá hacer preguntas de cómo usar el paquete. Si el personal de ventas parece entendido, es probable que le pueda ayudar cuando tenga dudas. Recuerde que es el distribuidor, y no el fabricante del software, el que se espera que le ayude.
- *Reputación del fabricante.* Algunas compañías de software tienen buena fama por escribir software libre de fallos y fácil de usar. Su distribuidor debe saber qué compañías tienen mejor software. Otras consideraciones importantes son cuánto tiempo hace que el programa está disponible y si el fabricante del software ayudará al distribuidor para que éste le ayude a usted.
- *Requerimientos del sistema.* Algunos programas requieren una gran cantidad de RAM, mientras otros requieren un monitor de color o un periférico especial. Asegúrese de que un programa funciona con la configuración de su computadora, antes de comprarlo, especialmente si tiene una computadora que no es IBM.
- *Manuales de software.* La documentación de un programa debe ser fácil de leer y debería explicar todos los problemas que usted pueda encontrar. Muchos fabricantes permiten comprar los manuales sin los programas, de forma que se pueden leer antes de comprar el software.
- *Cuánta formación está disponible.* Muchas tiendas y especialistas en formación han montado cursos para enseñar a utilizar los productos software más populares. Incluso algunas tiendas incluyen esta formación, sin coste adicional, cuando se compra el software. Muchos paquetes de software populares tienen programas «tutoriales» que desde la pantalla de la computadora le enseñan a utilizar el software.

Una vez que ha decidido qué programa comprar, necesita decidir dónde comprarlo. El método más simple es averiguar si el distribuidor de su computadora vende el software. En este caso, puede que quiera comprarlo allí. Este método, sin embargo, puede no ser el mejor; la tienda quizá venda el software

al precio oficial, mientras que en otros sitios pueden hacerle descuentos. También puede que su distribuidor no esté familiarizado con el producto o que no lo tenga en existencias. (A menudo, los distribuidores tardan semanas en recibir el software).

Tiene dos principales opciones si se decide a comprar: comprar el software en un distribuidor de computadoras o tienda especializada, o bien comprarlo en una compañía de venta por correo. Casi todas las tiendas de computadoras venden algo de software, aunque usualmente se limitan a unos cuantos programas de mucha venta. Las tiendas que se especializan en software suelen tener cientos de programas; son también una buena fuente de información para saber qué productos le hacen la competencia a un producto particular. Las compañías de venta por correo varían en calidad de unas a otras, pero tienen la ventaja de tener bajos precios y existencias en cantidad suficiente.

Naturalmente, hay pros y contras en cada una de las posibilidades. Los distribuidores al por menor suelen cobrar más por el software y no suelen tener tan amplia variedad de existencias como las compañías de venta por correo. Por otro lado, las tiendas al por menor, especialmente si son sólo de software, suelen dar mejor servicio a los clientes que las de venta por correo. En las tiendas puede tener la oportunidad de probar un número de productos competidores (como por ejemplo, unos cuantos procesadores de textos). Sin embargo, las compañías de venta por correo también tienen ventajas. Además de tener precios más bajos, algunas compañías de venta por correo tienen un servicio de clientes bueno, o incluso excelente (puesto que la mayor parte de las dudas sobre el software pueden resolverse por teléfono). La tabla 6-1 resume las ventajas y desventajas de estos tres tipos de tiendas.

Tabla 6-1. Comparación de las ventajas de los diferentes distribuidores de software

Consideraciones	Tienda de computadoras	Tienda de software	Venta por correo
Precio (en general)	Más alto	Intermedio	Más bajo
Servicio post-venta	Bueno, si están familiarizados con el produto	Mejor	Usualmente bastante pobre (usuele implicar una llamada de larga distancia)
Surtido de productos competidores	Regular (a menudo sólo tienen los de más venta)	Muy bueno, suele ser muy útil para decidir entre varios paquetes	Mejor, aunque no muy útil para elegir
Formación	Buena	Buena	Usualmente ninguna

Algunos programas no están disponibles ni en las tiendas al por menor ni en las compañías de venta por correo. Si un fabricante es demasiado poco importante para conseguir la distribución por los cauces normales, puede vender el producto por correo. Usualmente, la única forma de encontrar estos productos es leyendo anuncios en las revistas. Comprar software de esta manera es arriesgado, a no ser que haya leído críticas favorables o conozca a alguien que tenga el producto.

Si quiere pagar para poder probar un programa, quizá pueda alquilarlo en una compañía de alquiler de software. La tarifa de alquiler suele ser un porcentaje del coste del paquete y frecuentemente se le permite utilizarlo unas dos semanas. Si decide comprar el paquete en compañía de alquiler, la tarifa de alquiler se suele descontar del precio de venta.

A pesar de sus buenas intenciones, puede que compre un programa de aplicación que no le guste. A no ser que el fabricante de dicho paquete tenga esa política, es improbable que un distribuidor le permita devolver el software si usted no está satisfecho. Uno de los desafortunados hechos de la vida en la industria de las computadoras es que puede que tenga que comprar más de un paquete del mismo tipo particular de software, para conseguir el que es más adecuado para usted. Una compra cuidadosa puede reducir grandemente sus posibilidades de equivocación.

Software de dominio público y software compartido

El *software de dominio público* es el software que no está patentado y puede ser copiado libremente. Hay miles de programas de dominio público, muchos de los cuales son extremadamente útiles. Por ejemplo, algunos de estos programas ayudan a transferir ficheros entre paquetes de aplicación incompatibles (como por ejemplo, de un programa de hoja de cálculo a otro), enseñan cómo programar o son ejemplos de cómo usar BASIC para realizar tareas comunes. Hay también cientos de juegos gratuitos.

La forma más fácil de encontrar estos programas es hacerse socio de una asociación local de usuarios de computadoras. Hay grupos específicos para el IBM PC y similares. La mayoría de estas asociaciones tienen bibliotecarios con colecciones completas de software de dominio público. Estos suelen copiar los programas que usted necesita si trae sus propios disquetes o le venden disquetes con los programas ya copiados. Las revistas mencionadas en el capítulo 7 tienen frecuentemente listas de asociaciones locales de usuarios. Muchas tiendas de computadoras pueden proporcionarle también esta información.

El *software compartido* es una alternativa interesante al método tradicional de comprar software. Muchos de los programas de software compartido están bien escritos y son fáciles de usar; algunos son mucho mejores que los correspondientes de software normal o de software de dominio público. El software compartido es un software que se obtiene gratuitamente y que, si le

gusta, usted envía a sus creadores una contribución voluntaria. Puede obtener copias de software compartido de las asociaciones de usuarios o de amigos. Este concepto es radicalmente diferente de las ventas de software típicas. Cualquiera puede dar copias del software y usted no está obligado a pagar a los que lo desarrollaron. Sin embargo, si no paga por el software, no puede esperar asistencia por parte de los autores.

A menudo la gente que escribe software compartido incluye el código fuente (las instrucciones de programación) junto con el software. Si es usted un programador y mejora el producto, los autores del software le piden que les mande las mejoras, de forma que puedan compartirlas con otros. De esta manera, mucha gente contribuye a mejorar el código fuente, haciendo que los programas resultantes estén usualmente libres de fallos.

Aunque el concepto de software compartido es nuevo, está alcanzando gran popularidad. En este capítulo encontrará las descripciones de tres buenos programas de software compartido: PC-Write para procesamiento de textos, PC-FILE para gestión de base de datos y PC-TALK para comunicaciones. Si tiene un amigo que tenga estos programas, pídale copias. Si le gustan, envíe la contribución sugerida a los autores; si no, no habrá perdido nada en la operación.

SOFTWARE DE PROCESAMIENTO DE TEXTOS

El software de procesamiento de textos permite escribir y conservar cualquier tipo de texto (memorias, cartas, informes y libros) e imprimirlo en una impresora. Una vez que lo pruebe, pensará que el procesador de texto es una herramienta extremadamente valiosa en su computadora. Debería tener un buen procesador de texto, incluso si compró la computadora para ejecutar otros paquetes de aplicación.

Según se van haciendo más comunes los programas de procesamiento de texto, mucha gente se va dando cuenta de la enorme diferencia entre usar un procesador de texto y una máquina de escribir. Con un procesador de texto, el revisar un texto es extremadamente fácil, reestructurar una carta o una memoria es muy rápido y conservar texto en disco significa que nunca tendrá que volver a escribirlo. Esta primera sección cubre algunos fundamentos importantes para usar un procesador de texto.

Cómo trabaja un procesador de texto

Probablemente le resulta fácil imaginar un programa que visualiza los caracteres en la pantalla cuando los introduce por el teclado. Exceptuando la pantalla, el programa actúa igual que una máquina de escribir normal. También puede imaginar que cuando usted teclea los caracteres, el programa los guarda en un fichero; posteriormente usted puede dar una orden para impri-

mir el fichero en la impresora. Hasta ahora, esta descripción coincide con la de una máquina de escribir con memoria: una vez que se escribe algo, siempre se puede imprimir.

Sin embargo, la similaridad termina cuando se quiere modificar lo que se ha tecleado. Con un procesador de texto se pueden añadir o borrar palabras, frases o párrafos, fácilmente. También se pueden trasladar frases y párrafos por todo el fichero texto que se ha creado. Al hacer estos cambios, su programa le está permitiendo *editar* el texto. Otra característica de un procesador de texto es que se puede diseñar un documento para que tenga un formato particular (numeración de página automática, indentación, letra cursiva, etc.) cuando se imprima en un papel.

Exactamente igual que como se daban las órdenes del MS-DOS, en los primeros capítulos de este libro, se dan las órdenes de un procesador de texto, para decirle lo que se quiere hacer y dónde. Con la mayoría de los programas se utiliza una combinación de pulsaciones de teclas para dar estas órdenes (como por ejemplo, para moverse hacia adelante o hacia atrás en el texto, para borrar, para insertar, etc.). La mayoría de las computadoras MS-DOS tienen teclas especiales en el teclado que hacen más fácil el procesamiento de textos.

Las cartas, los capítulos o los informes que se escriben con un procesador de texto se guardan en ficheros como los tratados en el capítulo 2. Si está introduciendo un nuevo fichero, dígale al programa que quiere introducir texto y, a continuación comience a teclear. Si está editando texto que ya había escrito, dígale al programa el nombre del fichero que quiere editar y dónde está dentro del texto lo que usted quiere editar (como por ejemplo, la primera línea o la segunda frase del párrafo cuarto). Cuando el programa le lleve allí, podrá hacer los cambios. En realidad, usted se mueve de esta forma a lo largo de todo el fichero, cambiando el texto antiguo o añadiendo nuevo texto. Una de las características agradables de la edición es que se puede mover hacia atrás y hacia adelante por el fichero, a voluntad; no hay la limitación de ir sólo del principio al fin.

Debe guardar su trabajo en el disco periódicamente, mientras lo está escribiendo y, a continuación, guardarlo de nuevo cuando haya terminado de editar o de introducir el fichero. Si quiere, puede imprimir este nuevo fichero. Después de haber usado software de procesamiento de texto para editar texto, puede que no quiera volver a utilizar nunca una máquina de escribir.

Un buen programa procesador de texto (llamado *editor orientado a pantalla*) permite ver una pantalla llena de líneas. Esta característica es una de las razones principales por las que un procesador de texto le será útil, incluso aunque no lo use para procesar texto. Cuando quiera editar un fichero con la orden del MS-DOS EDLIN, sólo verá una línea del fichero cada vez; en comparación, los procesadores de texto permiten ver como mínimo 15 líneas, de forma que usted pueda evaluar la línea en el contexto que la rodea.

La mayoría de los procesadores de texto muestran cómo aparecerá el texto en la página. Tales programas se suelen denominar programas de «vea lo que

obtiene». Otros programas muestran sólo los contenidos del fichero en la pantalla. Con estos otros, no verá el formato de su texto hasta que imprima su fichero en la impresora. Se utilizan órdenes de formateado para decirle a la impresora cómo formatear el texto.

Generalmente, cuanto más potente es el procesador de texto, más cosas se pueden hacer con sus *órdenes de formateado.* Hay muchos tipos de órdenes de formateado. Algunas permiten hacer cosas típicas sencillas (como poner los márgenes izquierdo y derecho, numerar páginas o hacer una indentación en un párrafo), mientras que otras son más sofisticadas. Por ejemplo, tales órdenes pueden hacer cosas como una indentación en una lista automáticamente, centrar los títulos de las secciones, poner un encabezamiento específico en la cabecera de cada página, o poner en negrita o subrayar automáticamente un grupo de palabras.

Ambos métodos tienen ventajas: si quiere ver en la pantalla cómo quedará exactamente un documento, entonces los programas de «vea lo que obtiene» serán los mejores para usted. Sin embargo, esta característica limita el tipo de las órdenes de formateado que el software puede ofrecer. Ninguno de los dos es inherentemente mejor; son diferentes. Debe elegir qué tipo de programa usar, según sus necesidades.

¿Qué importancia tienen las órdenes de formateado? Para escribir documentos cortos, no se necesitan las características sofisticadas de formateado de la impresión, que tienen muchos procesadores de texto. Debe interesarse más en las posibilidades de edición. Por otro lado, cuanto más sofisticadas sean las órdenes de formateado mejor aspecto tendrán los informes y los documentos más largos. Si sus documentos son largos (por ejemplo, si escribe manuales o libros), las características de formateado más avanzadas podrían hacer su trabajo mucho más fácil. Por ejemplo, las características que numeran automáticamente las cabeceras o que generan un índice de materias son muy importantes en documentos largos.

Qué buscar en un procesador de texto

Hay una gama muy amplia de precios para los procesadores de texto disponibles. Los precios están incluidos generalmente en dos grupos: por debajo de 100 $ y entre 250 $ y 600 $. Curiosamente, no se puede asumir que un procesador de texto más caro, sea más fácil de usar y tenga más posibilidades que una versión más barata. Algunos procesadores de texto baratos tienen más funciones que otros que son caros. Los criterios más importantes son, primero, si un paquete es fácil de aprender y usar, y, segundo, si puede hacer todas las cosas que usted necesita que haga.

Si un procesador de texto es importante para su trabajo, debe hacer la selección cuidadosamente. Primeramente, valore a fondo sus necesidades; la mayor parte de los hombres de negocios utilizan procesadores de texto para escribir, editar y formatear cartas, memorias e informes. A continuación,

pruebe varios programas. La mejor forma de hacerlo es llevar una muestra de su trabajo a una tienda de venta al por menor. Si escribe un montón de cartas, pruebe a escribirlas utilizando varios programas diferentes. ¿Es fácil producir la carta exacta que usted quería? Debería tomar notas de cada programa y compararlas.

Las siguientes son algunas cuestiones a considerar cuando comience la búsqueda del mejor programa:

- ¿Es fácil introducir, cambiar, formatear e imprimir el texto? Probablemente usted quiere evitar programas que requieran memorizar secuencias complicadas de teclas para dar las órdenes de edición.
- ¿Puede el procesador de texto funcionar con su impresora? Algunas impresoras pueden cambiar de tipo de letra (denominado *fundición* en lenguaje tipográfico), usar espaciamiento proporcional o crear superíndices o subíndices. Sin embargo, sólo se pueden usar estas características si el procesador de texto conoce el tipo de impresora que usted tiene. La mayor parte de los procesadores de texto sólo pueden funcionar con unos cuantos tipos de impresoras; si el que usted compra no conoce su impresora, no usará las características especiales.
- ¿Incluye el programa una lista de correo? Muchos programas permiten usar un fichero de lista de correo con una carta para producir un envío, en serie, insertando el nombre y la dirección en la parte de arriba de cada carta.
- ¿Tiene el programa características para usuarios avanzados? Mucha gente que utiliza de forma regular un procesador de texto se vuelve muy hábil con el programa. Muchos paquetes tienen características avanzadas que, aunque demasiado complicadas para usuarios que empiezan, dan a los usuarios avanzados mucha más flexibilidad. Un ejemplo de estas características es la capacidad de ejecutar muchas órdenes al mismo tiempo.
- ¿Es fácil de aprender? Algunos programas vienen con una documentación bien escrita, menús de ayuda por pantalla y plantillas o tarjetas de referencia de las órdenes para ayudarle a aprender el sistema lo más pronto posible. Otros programas tienen documentación pobre, órdenes difíciles de teclear y otras características que le impiden sentirse a gusto con el programa.

La mejor manera de determinar si un procesador de texto es fácil de aprender, es de palabra. Si puede, pregúntele a alguien que haya aprendido a usar el procesador de texto, o pregúntele al distribuidor. Probablemente vale la pena que asista a una reunión de una asociación local de usuarios de computadoras y pregunte a otras personas cuál le recomendarían. Sin embargo, le advertimos que éste es un tema controvertido; preguntar a la gente qué procesador de texto comprar produce el mismo tipo de juicios subjetivos que el preguntar por las preferencias religiosas o por los coches favoritos.

Las revistas son otra buena fuente de información, especialmente si contienen críticas del programa en que está usted interesado. Estos artículos tienen frecuentemente diagramas comparativos que muestran las características de cada procesador de texto examinado. El capítulo 7 menciona algunos libros que dan introducciones completas al procesamiento de textos, además de revistas que contienen críticas de productos. Si no tiene experiencia, debe seleccionar un procesador de texto fácil de usar, en vez de uno con más características. Sin embargo, si ya ha aprendido a manejar un procesador de texto, aprender a manejar otro le resultaría bastante fácil.

Debe asegurarse de que el paquete que compra es un procesador de texto y no simplemente un editor de texto (a no ser que usted quiera esto último). Hay una gran diferencia entre los dos: los editores de texto no formatean el texto y se usan por los programadores (puesto que los programas no necesitan formateado) o por gente que tiene un paquete separado de formateado de textos. La mayor parte de los procesadores de texto tratados en esta sección tienen ambas capacidades de edición y formateado.

Unos cuantos paquetes de procesamiento de textos incluyen programas de *verificación ortográfica.* Un verificador ortográfico examina cada palabra de su fichero texto, la busca en el diccionario y le dice cuáles no reconoce. A continuación, le permite decidir si usted escribió mal la palabra o si el programa no está familiarizado con la palabra (por ejemplo, siglas y nombres propios). Como puede imaginar, cada fichero podría tener muchas palabras que no están en el diccionario del programa; por ello, es especialmente importante conseguir un programa de verificación ortográfica que permita actualizar fácilmente el diccionario. Estos programas se describen más adelante en esta sección.

En el resto de esta sección se presentará un muestreo de programas procesadores de texto, editores de texto y formateadores, y verificadores ortográficos, que están actualmente a la venta. Se comenzará por ver las alternativas disponibles si no se desea comprar un programa de procesamiento de texto.

Programas disponibles gratuitamente

El MS-DOS trae un editor de texto poco útil llamado EDLIN. Francamente, es difícil tener en cuenta a EDLIN si se quieren escribir o editar más de unas cuantas líneas de texto. Se puede usar, pero es difícil de aprender y hace extremadamente poco por usted. Si el fabricante de su computadora le incluyó junto con ésta un paquete de procesamiento de texto, probablemente ignorará por completo EDLIN.

La principal dificultad al utilizar EDLIN es que es un editor de línea; esto es, no le permite ver una pantalla llena de información, sino sólo una línea. De este modo, usted no se puede mover por el fichero fácilmente, puesto que EDLIN siempre se mueve hasta el comienzo de la línea. EDLIN tampoco tiene capacidad de formateado de texto.

Si no puede gastarse el dinero en comprar un sistema profesional de procesamiento de texto (algunos cuestan sólo 50 $), hay una bastante buena alternativa gratuita a EDLIN. *PC-Write,* un paquete de software compartido de Quicksoft, tiene muchas de las características que se encuentran en los programas que cuestan de 200 $ a 500 $ (vea la discusión de los programas de software compartido, previamente en este capítulo). La contribución solicitada para el PC-Write es de 75 $, por la cual usted recibe una copia impresa del manual, el programa fuente (escrito en Pascal) y un número de teléfono por si necesita ayuda.

El PC-Write tiene unas capacidades de edición y formateado bastante buenas. Puede usarlo para editar cualquier fichero texto, e incluye capacidades limitadas de formateado de la impresión. El editor utiliza las teclas de función del IBM PC en vez de las teclas de control, para editar texto. Tiene una interesante serie de características avanzadas, incluyendo la edición con pantalla dividida. La documentación, que está incluida en el disco, no está bien escrita, pero puede ciertamente aprender lo básico, leyendo los primeros capítulos.

Hay otros procesadores de texto de dominio público, pero ninguno iguala al PC-Write en capacidades ni en lo completo. Incluso el peor de ellos suele ser mejor que EDLIN.

Programas de procesamiento de texto a la venta

Esta sección proporciona una amplia visión general del mercado, describiendo unos cuantos de los programas disponibles. Hay más de 50 paquetes de procesamiento de textos para el MS-DOS, y cada mes aparecen otros nuevos. Como resultado, es muy probable que pueda encontrar alguno que satisfaga fielmente sus necesidades.

Uno de los paquetes más fáciles de usar es *Volkswriter,* de Lifetree Software. En vez de un largo manual describiendo cómo usar el editor, la mayoría de las instrucciones se dan en unos cuantos ficheros. Para aprender el sistema, usted lee los ficheros y prueba cada orden de edición cuando se le presenta. El formateado de texto es, desgraciadamente, inadecuado para cuando se trata de algo más que cartas. Sin embargo, el paquete es muy popular entre los principiantes porque no es caro, tiene muchas características de edición y proporciona una buena introducción a la utilización de los procesadores de texto. *Volkswriter deluxe* tiene más características para usuarios avanzados, pero cuesta más.

Otro paquete fácil de aprender es *SuperWriter,* de Sorcim. El programa trae instrucciones completas en un manual, además de un manual suplementario denominado *10 Minutes to SuperWriter* que enseña a crear, editar e imprimir una carta en 10 minutos. Aunque algunas de las peticiones de órdenes no son muy claras, una pantalla de ayuda está disponible para cada petición de orden. Aquellos que han usado el popular programa de hoja de cálculo SuperCalc de Sorcim, reconocerán la mayoría de las peticiones de órde-

nes. Otra característica muy útil de SuperWriter es la historia del documento, que se mantiene al tanto de quién creó un documento y cuándo fue actualizado por última vez. Esto puede ser muy importante en una oficina donde mucha gente trabaja en un documento.

WordVisión, de Bruce & James Program Publishers, es un paquete de procesamiento de textos mucho más completo que Volkswriter o SuperWriter. El manual es muy fácil de seguir, aunque lleva bastante tiempo para empezar algún procesamiento de texto útil. El paquete incluye pequeñas etiquetas autoadhesivas que se supone que usted pega en las teclas del teclado; desafortunadamente, estas etiquetas hacen que el teclado parezca atestado. También puede pedir capuchones para las teclas, que son semipermanentes y más fáciles de usar. Aunque muchos programas pretenden mostrarle en la pantalla el aspecto que tendrá exactamente la salida, WordVision lo hace mejor que ningún otro. Su uso del color también hace que ciertas características sean más fáciles de ver. Por ejemplo, el texto en negrita se destaca del texto normal usando un color diferente en la pantalla (en monitores monocromos, esto se visualiza como un nivel diferente de luminosidad).

Si decide que no necesita ver el documento formateado inmediatamente, *The Final Word,* de Mark of the Unicorn, es un paquete excelente. Las órdenes de edición de textos son más fáciles de recordar que las de cualquier otro paquete y las posibilidades de la memoria ampliada del MS-DOS se utilizan en su más completa extensión. Es bastante fácil usar las órdenes de formateado para hacer informes avanzados y manuales, pero la documentación de las órdenes de formateado no es muy explícita. *Perfect Writer,* de Perfect Software, tiene muchas de las características de The Final Word, pero son más fáciles de usar en The Final Word.

PFS:Write, de Software Publishing Corp., tiene una ventaja interesante sobre otros muchos paquetes: permite mezclar con el texto gráfico de negocios del programa PFS:Graph. Los gráficos pueden hacer que la salida sea mucho más interesante y pueden facilitar la producción de informes. PFS:Write tiene sin embargo, una desafortunada limitación. Cada fichero que se edita tiene que ser bastante pequeño. Esto obliga a unir muchos ficheros pequeños si se está escribiendo un documento largo.

PeachText 5000, de Peachtree Software, es un procesador de texto que permite formatear texto para impresión, pero no muestra el formato de la página por pantalla. Muchos usuarios creen ciegamente en PeachText, mientras que otros piensan que el programa hace frecuentemente suposiciones incorrectas de lo que ellos quieren. Por ejemplo, la mayor parte de las medidas de seguridad del paquete requieren que se den dos grupos de pulsaciones de teclas, uno para dar la orden de edición y otro para confirmar que se quiere ejecutar la orden. Además, el manual no proporciona explicaciones completas de cómo usar las características de formateado. Esto las hace muy difíciles de descifrar.

MultiMate, de Softword Systems, hace que su computadora MS-DOS parezca un procesador de texto avanzado. En realidad, si está familiarizado con

los procesadores de texto de Wang, se sentirá muy cómodo con MultiMate. Características como las líneas de formato global (áreas de texto que describen el formato del documento) son idénticas a las del procesador de texto Wang.

WordPerfect, de Satellite Software International, tiene una multitud de características de edición y formateado. Es capaz de editar dos documentos a la vez, pero no en la misma pantalla. El formateador permite poner múltiples columnas de texto en la página y tener muchas líneas en los encabezamientos y pies de páginas. Este paquete es totalmente completo y, aunque el manual no está descrito para el principiante, usted sería capaz de usar muchas de las órdenes inmediatamente.

Benchmark, de Metasoft, es uno de los paquetes más difíciles de aprender. Usted se ve obligado a pasar por una gran cantidad de preparativos antes de poder siquiera ejecutar el programa. El gran número de opciones técnicas que se le presentan durante la instalación, puede intimidar a los principiantes. El programa tiene algunas características interesantes, como una calculadora interna, que no se encuentran en la mayoría de los procesadores de texto. Sin embargo, la documentación malamente escrita puede hacer difícil el aprovechar estas características.

Algunos programas le hartan a base de mensajes de ayuda y pantallas informativas. *VisiWord,* de VisiCorp, es uno de ellos. Los mensajes por pantalla y el manual bien escrito son importantes cuando se está aprendiendo y proporcionan una asistencia suficiente para los usuarios poco frecuentes. Pero una vez que usted aprende el sistema, puede que piense que los mensajes le estorban. Una gran ventaja de VisiWord sobre otros muchos procesadores de texto es la facilidad con que puede usarse para formatear tablas con números.

Microsoft Word, o simplemente *Word,* es probablemente el sistema de procesamiento de textos más potente disponible para el MS-DOS. Es fácil de aprender, pero puede que se sienta abrumado por el número de características avanzadas. Word le permite usar un *ratón,* que es un dispositivo controlado con la mano, que se usa para mover una flecha por la pantalla. El ratón apunta a las distintas opciones en el menú de órdenes, a una palabra (o grupo de palabras) para trasladar o borrar, o simplemente a la posición donde quiere comenzar a introducir texto. Algunos mecanógrafos se quejan de que usar un ratón es molesto, puesto que la mano tiene que dejar el teclado para moverlo. Sin embargo, se puede usar Word sin el ratón.

Microsoft Word utiliza «glosarios», que son colecciones de reglas relativas al texto que usted guarda en disco. Por ejemplo, puede decirle a Word que cada vez que vea un título, debe centrar la línea e imprimirla con caracteres en negrita. A continuación, lo único que necesita hacer es marcar una línea como título y Word la formateará. En realidad, se pueden tener muchos glosarios con reglas diferentes. Hay otras muchas características avanzadas que hacen de Microsoft Word una excelente opción para un procesador de texto con todo tipo de funciones.

Textra, de Ann Arbor Software, es uno de los mejores paquetes de procesamiento de texto a bajo precio. Tiene muchas de las características de los

paquetes más grandes (como por ejemplo, ver anticipadamente el texto formateado) y un programa tutorial muy bueno que enseña a manejar el producto, de una forma muy simple.

WordStar, de MicroPro, es uno de los programas de más venta. Fue uno de los primeros editores de pantalla para microcomputadoras. Muchos distribuidores de software piensan que es fácil de vender debido al fácil reconocimiento de su nombre. Sin embargo, no es tan fácil de usar como muchos de los otros programas mencionados aquí. Sus órdenes son fáciles de olvidar y algunas personas que lo usan a diario piensan que todavía necesitan tener a mano una relación de las órdenes.

Mirándolo por el lado positivo, hay más de una docena de libros disponibles para aprender WordStar y hay más cursos de formación para el WordStar que para cualquier otro programa de procesamiento de texto. Con WordStar todas las opciones de edición en cada momento se visualizan en la pantalla. Algunos encuentran que esta característica no es conveniente porque el menú quita casi un tercio del área de texto utilizable. Si quiere «el estándar», WordStar lo es. Recuerde, sin embargo, que hay muchos otros programas y algunos de ellos le pueden ir mejor para sus propósitos, pueden ser más fáciles de aprender y menos caros.

Otros procesadores de texto populares para el MS-DOS son *Palantir,* de Designer Software, *WordPlus* de Professional Software, *Easy Writer,* de Information Unlimited Software, *XyWrite II,* de Xyquest, *ReadiWriter,* de ReadiWare Systems y *Blue,* de Symmetric Software.

Editores de texto y formateadores

Aunque hay una amplia selección de procesadores de texto, puede que por sus especiales circunstancias quiera usar un editor de texto y un formateador, separadamente. Estos editores y formateadores especializados tienen frecuentemente características destacadas que no están disponibles en los procesadores de texto. (Recuerde que los procesadores de texto tienen un editor de texto y un formateador combinados en un único programa).

Para usuarios avanzados, *VEDIT,* de CompuView, es un editor de texto excelente. Se puede programar VEDIT para que realice un grupo de tareas de edición en una sola orden. Por ejemplo, le puede decir a VEDIT que busque la siguiente aparición de una palabra y la cambie de una determinada manera si está al final de una frase y de otra manera si no lo está. Hay muchas características de VEDIT que se parecen a TECO, un potente editor que funciona en computadoras de gran tamaño.

Otro buen editor es *MINCE,* de Mark of the Unicorn Company (los mismos de The Final Word). MINCE tiene un cierto número de características de edición que no están disponibles en ningún otro, y esto permite trabajar con ficheros muy largos. En realidad, MINCE se utilizó para escribir la mayor

parte de este libro. Mark of the Unicorn también tiene un formatedor de texto llamado *SCRIBBLE.*

Los formateadores de texto permiten formatear el texto que se ha editado con algún editor de texto. Un formateador muy bueno que funciona con cualquier editor de texto es *MicroScript,* de MicroType. Tiene la mayor parte de las características de otros formateadores y características adicionales para texto en multicolumna, listas, notas a pie de página y generación automática del índice de materias y del índice. El programa puede usar las órdenes de impresión que están disponibles en una amplia variedad de impresoras.

Otros buenos editores son *Edit Tool,* de Amerisoft, *PMATE,* de Phoenix Software Associates, *Edix,* de Emerging Technology, y *KEDIT,* de Mansfield Software Group. Otros buenos formateadores son *ProScript,* de SoftCraft y *TypeGraph,* de Soft Lab.

Verificadores ortográficos

La mayor parte de los programas de verificación ortográfica operan de la misma manera: usted los ejecuta, ellos extraen las palabras que no reconocen y usted dice si hay que añadir la palabra al diccionario del programa o hay que corregirla. Estos programas pueden ahorrar una gran cantidad de tiempo a todos aquellos que no dominan la ortografía. Incluso si usted escribe correctamente, estos programas le ayudarán a corregir errores de mecanografía, porque detectan todo lo que no está en sus diccionarios.

Cuando esté evaluando verificadores ortográficos, los factores más importantes son la facilidad de uso y su exactitud. Un gran número de palabras en el diccionario no suele ser importante si el programa permite añadir palabras fácilmente. La mayor parte de la gente usa menos de 5.000 palabras en todos sus escritos y casi todos los verificadores ortográficos disponibles tienen al menos 25.000 palabras en su diccionario. Algunos verificadores ortográficos pueden ejecutarse con el procesador de texto, de manera que se puede editar y verificar la ortografía fácilmente.

The WORD Plus, de Oasis Systems, es un verificador ortográfico muy bueno. Es fácil de usar y tiene algunas características originales. Cuando el programa encuentra una palabra que no reconoce, le muestra la palabra; si usted piensa que la ortografía correcta se parece a la que había dado, el programa buscará palabras similares en el diccionario. Así, puede utilizarlo para que le ayude a encontrar la palabra correcta sin saber la ortografía correcta.

Otro buen verificador ortográfico es *MicroSpell,* de Trigram Systems. Igual que The WORD Plus, MicroSpell busca palabras similares, pero puede que con bastante frecuencia no encuentre la correcta. Esto es debido a que utiliza un método de verificación denominado *procesamiento de la raíz,* que asume que los prefijos y sufijos corrientes añadidos a una palabra son siempre correc-

tos. Así, un verificador ortográfico de este tipo diría que «untree» y «radioer» son palabras válidas (1).

HOJAS DE CALCULO

A los programas de hoja de cálculo, también llamados de hoja electrónica, se les puede atribuir gran parte del mérito de hacer populares las microcomputadoras en las pequeñas empresas. Hasta que se lanzó VisiCalc, de VisiCorp, el uso de las computadoras personales en los negocios estaba casi exclusivamente limitado al procesamiento de textos. (La gestión de base de datos, tratada más adelante en este capítulo, no se había introducido todavía, y el software de negocios acababa de empezar a desarrollarse). La microcomputadora se convirtió en una herramienta mucho más viable para los negocios cuando se desarrolló un programa para modelar el crecimiento de una compañía y coordinar los ingresos y gastos. Esto hizo que la computadora se pareciera mucho más al papel cuadriculado de un contable.

La idea básica que hay detrás de un paquete de hoja de cálculo es muy simple: la pantalla se divide en *celdillas* rectangulares, similares a las cuadrículas de la hoja de balance de un contable. La figura 6-2 muestra un ejemplo de hoja de cálculo. Las columnas están marcadas con letras y las filas con números. Las celdillas se nombran por la columna y la fila en que están. Por ejemplo, el número 110201 en la celdilla B4.

Cada celdilla puede contener uno de los tres tipos de información siguientes: números, ecuaciones o texto.

- *Números.* Pueden representar el número de dólares ganado por un departamento en un año, el número de artículos en un inventario, la cantidad de tiempo gastado en un proyecto, o virtualmente cualquier otro número. Estos números están fijados; esto es, son datos que usted introduce. Por ejemplo, en la figura 6-2, el número que está en la celdilla B4 representa el número de dólares ganado en ventas de hardware.
- *Ecuaciones.* Las ecuaciones relacionan las celdillas entre sí. Por ejemplo, la celdilla B13 de la figura 6-2 contiene la ecuación B4+B7+B10. Sumando los contenidos de esas tres celdas se obtienen los ingresos totales en 1981. De la misma manera, las celdillas de la columna F contienen ecuaciones que proyectan las ventas probables para 1985 sobre la base del crecimiento de las ventas en los años anteriores. Las ecuaciones

(1) *N. del T.:* Como puede apreciarse, la verificación ortográfica está orientada hacia la lengua inglesa.

	A	B	C	D	E	F
1		1981	1982	1983	1984	1985
2						estim.
3						
4	Ventas	110.201	124.933	121.640	145.283	148.910
5	Hardware					
6						
7	Ventas	48.270	65.391	86.922	94.280	97.640
8	Software					
9						
10	Alquileres	0	0	15.900	32.740	46.000
11						
12						
13	Ingresos	158.471	190.324	224.462	272.303	292.550
14	totales					

Figura 6-2. Un ejemplo de hoja de cálculo que muestra números y rótulos (no se muestran las ecuaciones)

no se visualizan en la hoja de cálculo, aunque puede verlas cuando lleve el cursor a la celdilla en particular. Si no, sólo se visualizan los resultados de la ecuación.

Las ecuaciones pueden utilizar muchas funciones matemáticas. Por ejemplo, una celdilla puede contener una ecuación que determine qué porcentaje de las ganancias de la compañía fueron aportadas por un departamento individual.

- *Texto.* El texto se utiliza normalmente en la hoja de cálculo en rótulos que explican lo que los números representan. Un ejemplo es la palabra Alquileres en la celdilla A10.

Lo maravilloso de los programas de hoja de cálculo es que permiten cambiar los números y las ecuaciones y, a continuación, ver los resultados. (Esto se llama *cálculo de «qué-pasa-si»*). Por ejemplo, si quisiera ver qué ocurriría si el modelo de ventas mostrado en la figura 6-2 se desplazara para concentrarse más en los alquileres, cambiaría los números de la columna 1984 y vería cómo afectaría esto a la columna 1985. O bien, podría cambiar las ecuaciones que utilizó en la columna 1985 y ver cómo cambiarían las ganancias previstas.

Los programas de hoja de cálculo tienen numerosas aplicaciones como herramientas de propósito general para modelar con números. Aparte de darle un método fácil para hacer estimaciones sobre el futuro, también proporcionan un método sofisticado para investigar el «cash flow» (estimación de teso-

rería) de una compañía. Después de mirar las distintas formas por las que el dinero entra y sale de una compañía, un gerente puede examinar las posibles estrategias para maximizar los beneficios. Las hojas de cálculo también se usan para modelar problemas de ingeniería o para casi cualquier otra tarea que requiera ecuaciones.

La mayor parte de las hojas de cálculo realizan tareas idénticas. Las diferencias principales entre ellas son la facilidad con que se pueden aprender y usar, y el número de características adicionales que proporcionan. Algunas hojas de cálculo pueden usar más memoria de su computadora que otras y así darle más celdillas para usar. Prácticamente todos los paquetes de software integrado contienen un programa de hoja de cálculo.

Es difícil juzgar qué características hacen que merezca la pena comprar un paquete de hoja de cálculo. La capacidad de ordenar filas o columnas es la diferencia más interesante entre paquetes, aunque los anuncios de los paquetes con esta característica exageran su utilidad. (Ordenar es útil si tiene una larga lista de números y quiere ver rápidamente cuáles son los mayores y los menores). Usted debería comprar una hoja de cálculo basándose en si es fácil de aprender y de usar. El hecho de que muchos distribuidores ofrezcan clases de formación, para hojas de cálculo particulares, es probablemente una razón suficientemente buena para elegir un fabricante en vez de otro.

De los paquetes de hoja de cálculo que no están integrados con otro tipo de software, *VisiCalc* es el más popular. *Multiplan,* de Microsoft, es similar. Tiene una mejor interfaz de usuario y el manual proporciona muchos ejemplos útiles. *MicroPlan,* de Chang Laboratories, tiene muchas características avanzadas, incluyendo un modo de programación. *CalcStar,* de MicroPro, tiene la ventaja de utilizar órdenes similares a las de WordStar de forma que usted puede acostumbrarse a él bastante rápidamente si está familiarizado con WordStar. Hay docenas de otros programas de hoja de cálculo para el MS-DOS a la venta. Muchos de ellos tienen nombres casi indistinguibles, y muchos incluyen «calc» en alguna parte del nombre. Asegúrese de que compra el que usted pretendía comprar.

SISTEMAS DE GESTION DE BASES DE DATOS

Un *sistema de gestión de bases de datos (DBMS)* es un programa que organiza información, extrae información seleccionada, permite añadir nueva información y actualiza la información que ya está en la computadora. Un DBMS puede ayudar a mantenerse al tanto del inventario, libro mayor, nombres de clientes y direcciones, etc. También puede usar el DBMS para sacar información de su base de datos (como, por ejemplo, una lista de todas las cuentas vencidas y no pagadas, ordenadas por el importe de la deuda).

Cómo funciona un DBMS

Una *base de datos* es una colección de informaciones que están almacenadas en la computadora. Cada base de datos tiene una *estructura,* que es el modelo básico o formato en que se guarda la información. La estructura define qué tipos de información se pueden almacenar (nombres, direcciones, números de artículos, etc.), y si la información es texto o valores numéricos. Por ejemplo, si tiene una base de datos de clientes, la estructura de los datos podría parecerse a la tabla 6-2.

La información que hay en la estructura está contenida en *campos* y *registros.* Los campos son partes de la información que tienen un propósito particular en el conjunto de datos. Por ejemplo, una entrada para un nombre es un campo, una dirección es un campo, y así sucesivamente. La estructura define las características de cada campo.

Un grupo de campos relacionados se denomina registro y cada registro tiene un conjunto de valores para sus campos. Una buena analogía es un fichero en una biblioteca. El fichero en sí es el conjunto de datos, cada ficha es un registro y cada parte similar de información (como el nombre del autor o el título del libro) es un campo.

La figura 6-3 muestra un registro de la base de datos definida en la tabla 6-2. Cada campo aparece en el registro exactamente una vez. No necesita llenar todos los campos de un registro. La figura 6-4 muestra una vista esquemática del aspecto de una base de datos. Como puede ver, cada registro se ajusta a la organización global de la base de datos.

Tabla 6-2. Estructura de una base de datos de clientes

Campo	Tipo	Notas
N.º de cliente	Número	Siempre 4 dígitos
Compañía	Texto	Máximo 20 caracteres
Nombre de contacto	Texto	Máximo 25 caracteres
Dirección	Texto	Máximo 25 caracteres
Ciudad	Texto	Máximo 15 caracteres
Estado	Texto	Máximo 2 caracteres
Código postal	Número	Siempre 5 dígitos
Teléfono	Texto	Siempre en la forma ###-###-####
Nombre del representante de ventas	Texto	Máximo 25 caracteres
Saldo	Número	Presentado en los informes en dolares

N.º de cliente: 4120
Compañía: Phil's Stationery
Nombre de contacto: Tim Schmidt
Dirección: 2311 W. Palm Ave.
Ciudad: Ashville
Estado: CA
Código postal: 94766
Teléfono: 415-555-8110
Nombre del representante de ventas: Dale Byers
Saldo: $45193.12

Figura 6-3. Registro de una base de datos de clientes

El DBMS guarda la base de datos en un fichero del disco. Se usa el DBMS para cambiar registros o añadir nuevos registros a la base de datos. Por ejemplo, suponga que Phil's Stationery cambia su número de teléfono. Usted usaría su DBMS para encontrar el registro apropiado y, a continuación, cambiar el número de teléfono que aparece en el fichero.

Cambiar y añadir registros está muy bien, pero la principal razón por la que se usa un DBMS es para sacar información de la base de datos. Un buen DBMS permite ver la información basándose en prácticamente cualquier criterio que usted ponga. Algunas de las peticiones que podría hacer a su DBMS son

- Listar todas las compañías que tienen un código postal que empiece con 902.
- Escribir en la impresora todas las etiquetas de envío por correo para todos los clientes de Sharon Thompson.
- Listar todos los clientes que le deben dinero, ordenados por la cantidad que deben y mostrando el importe total de la deuda.

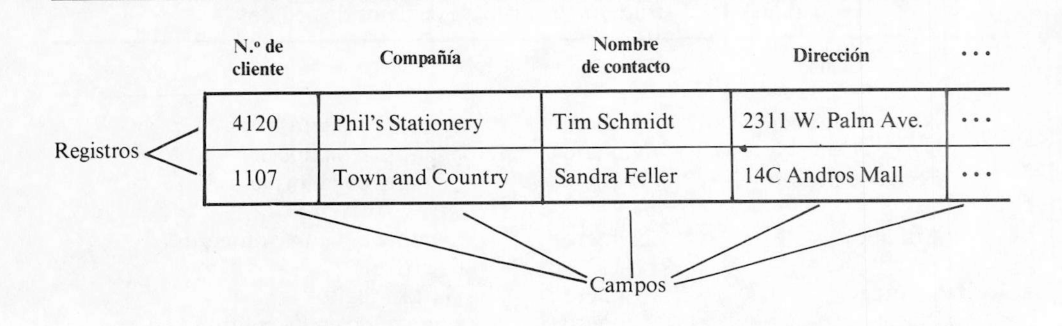


Figura 6-4. Vista esquemática de una base de datos

- Listar todos los clientes que se han añadido a la base de datos en el último mes.

Un DBMS es muy útil para hacer un análisis de la información en el acto, como por ejemplo, dar totales para las cantidades numéricas. Otra tarea que puede realizar bien es la de cribar la información. Por ejemplo, puede que usted necesite buscar un registro de una compañía, pero todo lo que recuerda de ella es que su nombre comienza con la letra P y está ubicada en San Francisco. Su DBMS puede encontrar simultáneamente registros que encajan en estos dos criterios.

Puede imaginar que a un DBMS le podría llevar mucho tiempo buscar información específica en un fichero de datos muy grande. Para acelerar la búsqueda, casi todos los DBMS a la venta utilizan *campos clave*. Cuando usted define una estructura, le dice al DBMS qué campos cree que usará normalmente para la búsqueda (en el ejemplo, nombre de la compañía y ubicación son buenas suposiciones). El DBMS, a continuación, guarda esta información en un fichero especial de claves que utiliza para encontrar los registros rápidamente.

Una característica muy pregonada de algunos sistemas se llama *procesamiento relacional.* Esta característica permite que un DBMS relacione dos bases de datos entre sí, según campos específicos. Esto es útil porque es normal tener más de una base de datos en la computadora, cada una con una estructura distinta, aunque ciertos campos sean los mismos. Un verdadero sistema relacional permite relacionar muchos ficheros de una vez.

Para ver cómo funciona el procesamiento relacional vea la tabla 6-3. Muestra una base de datos de los pedidos de una compañía. Observe que la estructura tiene un número de cliente de la misma forma que la de la tabla 6-2.

Con el procesamiento relacional, usted podría encontrar el nombre de todos los clientes que pidieron el artículo 33100 este mes, aunque ninguno de los dos conjuntos de datos contenían toda la información necesaria. Primero pediría a la base de datos de pedidos que encontrase todos los pedidos para el

Tabla 6-3. Estructura de una base de datos de pedidos

Campo	Tipo	Notas
Número de pedido	Número	Siempre 6 dígitos
N.º de cliente	Número	Siempre 4 dígitos
Fecha de pedido	Texto	Siempre en la forma ##/##/##
Artículo	Número	Siempre 5 dígitos; número de artículo
Cantidad	Número	Hasta 4 dígitos
Precio	Número	Presentado en los informes en dólares

número de artículo 33100, hechos este mes, y, a continuación, utilizaría la base de datos de clientes para listar todos los clientes cuyos números fueron encontrados en la primera petición. Esta relación entre los dos conjuntos de datos se muestra en la figura 6-5.

Clientes:

N. de cliente	Compañía	Nombre de contacto	Dirección	...
4120	Phil's Stationery	Tim Schmidt	2311 W. Palm Ave.	...
1107	Town and Country	Sandra Feller	14C Andros Mall	...

Pedidos:

Número de pedido	N.º de cliente	Fecha de pedido	Artículo	Cantidad	Precio
451552	1107	07/12/84	33011	144	422.00
451553	2832	07/12/84	18000	1	7.12

Figura 6-5. Relación entre los conjuntos de datos de clientes y pedidos

Hay muchas otras áreas en las que un DBMS relacional puede hacer la vida mucho más fácil. Por ejemplo, se pueden contestar fácilmente preguntas como «De las compañías que deben más de 2.500 $, ¿cuáles tienen pedidos por pagar?», incluso si la base de datos de pedidos está separada de la base de datos de contabilidad.

Ahora, puede que se le hayan ocurrido muchas otras formas de usar un DBMS. Quizá le interese saber si son fáciles de usar. Desgraciadamente, lleva por lo menos cinco horas aprender a usar un buen sistema de gestión de base de datos. Los más fáciles suelen tener limitaciones en su utilidad.

Lo que hay que buscar en un DBMS

Igual que hay docenas de procesadores de texto a la venta, hay también docenas de DBMS, muchos de los cuales tienen características similares. Como en el caso de los procesadores de texto, cada DBMS se suele anunciar como más fácil de aprender y más potente que sus competidores. Sin embargo, cualquier DBMS es inherentemente más difícil de usar que un procesador de texto; en realidad, la mayoría de la gente los usa menos de lo que se esperaba porque son difíciles.

Frecuentemente, utilizar un DBMS es igual que utilizar un lenguaje de programación que disponga de campos, registros y ficheros. Esto significa que cualquiera que vaya a crear sus propias bases de datos y aplicaciones, debería tener algún conocimiento de programación o querer aprender. No sería necesaria una gran cantidad de aprendizaje para crear una base de datos. En realidad, un manual razonable debería proporcionarle todo lo que usted necesita. Algunos DBMS no requieren que se use ninguna programación, en absoluto; en cambio, se usan menús directos para introducir los datos y producir informes.

El principal criterio para elegir un DBMS es para qué pretende usted utilizarlo. Si usa muchas bases de datos grandes y escribe informes complicados, debería comprar un DBMS avanzado y contar con pasar una gran cantidad de tiempo (al menos 40 horas) aprendiendo el sistema y escribiendo programas del DBMS para su aplicación. Sin embargo, recuerde que cuanto más complicado es el sistema de base de datos, más probable es que se arrincone en una estantería, cubriéndose de polvo. Por otro lado, si sus necesidades son bastante simples, un DBMS que no requiera programación probablemente se adaptará bien a sus propósitos.

Si realiza bastantes funciones estándar de contabilidad (como libro mayor o inventario), probablemente debería comprar un paquete de contabilidad. Las aplicaciones requeridas por la mayor parte de los negocios pequeños suelen estar disponibles en paquetes contables preprogramados. Las ventajas de estos paquetes son que usted no tiene que programarlos y que traen manuales completos, de forma que usted no necesita explicar a nadie más cómo usar su sistema. Los paqutes de contabilidad se tratan más adelante en este capítulo.

La mayor parte de los demás criterios que son importantes para elegir un DBMS (como la velocidad de procesamiento y la capacidad de manejar grandes ficheros) son muy difíciles de juzgar. Haga usted mismo sus propias averiguaciones, puesto que es excepcionalmente difícil tomar una buena decisión basándose en la publicidad. La mayoría de los DBMS más importantes han sido revisados en las revistas que se mencionan en el capítulo 7. La mejor forma de elegir un DBMS es preguntar a alguien en quien confíe cuál es el que usa, y probar unos cuantos programas con ese sistema.

Algunas de las restantes características importantes que se deben tener en cuenta antes de comprar un DBMS son

- Facilidad de creación de una pantalla de entrada de datos.
- Facilidad de obtención de informes.
- Características relacionales (como, por ejemplo, cuántos ficheros se pueden relacionar y si es fácil especificar las relaciones).
- Acceso por campo clave.
- Ayuda «on-line».

A menudo, es difícil comparar estas características, así que es importante conseguir el asesoramiento de usuarios con experiencia.

Si es principalmente, la cantidad de información y ayuda disponibles son muy importantes, puesto que estos programas son significativamente más difíciles de usar que la mayor parte de otros tipos de software. Desgraciadamente, hay muy pocas fuentes para este tipo de ayuda. Pruebe en las tiendas de computadoras y asociaciones de usuarios locales. Algunos consultores informáticos ofrecen clases informales y sesiones de ayuda. La disponibilidad de tales servicios puede ser un factor importante en la decisión de comprar un DBMS particular.

Algunos paquetes DBMS

Esta sección destaca unos cuantos tipos diferentes de programas DBMS de forma que pueda determinar lo que es importante para usted. Recuerde que sus mejores fuentes de información son otros usuarios del MS-DOS y su tienda de computadoras.

El DBMS más conocido en el mercado es *dBase II,* de Ashton-Tate. Al igual que WordStar, este paquete es popular a pesar del hecho de que muchos de sus competidores son más fáciles de usar, más rápidos y tienen más características. Sin embargo, puesto que dBase II es tan popular en el MS-DOS (al igual que en CP/M), la mayor parte de las tiendas de computadoras están más familiarizadas con él que con otros.

dBase II es más útil para programadores que están escribiendo aplicaciones de DBMS. Aunque Ashton-Tate intenta vender el programa a usuarios principiantes, es importante tener algún conocimiento de programación para sacar mayor provecho del software. Otras limitaciones son un número limitado de áreas de almacenamiento interno (lo cual dificulta la programación) y una relativamente baja velocidad de procesamiento. dBase II se anuncia como un DBMS relacional. Observe, sin embargo, que sólo puede relacionar dos ficheros a la vez, mientras que otros programas permiten relacionar cualquier número de ficheros.

Ashton-Tate ha trabajado duramente para mejorar la documentación de dBase II, pero sólo ha tenido un éxito parcial. El manual es bueno para programadores, pero puede ser difícil para principiantes. dBase II trae un programa tutorial, aunque el tutorial dejará rápidamente atrás a los principiantes.

R:base Series 4000, de Microrim, es similar a dBase II en su orientación hacia programadores, pero tiene muchas ventajas. Funciona significativamente más rápido, tiene una interfaz de usuario mejor, con muchos menús y pantallas de explicación, y sus manuales son más fáciles de entender. R:base proporciona una amplia ayuda «on-line» y ayuda a un programador principiante a través de muchas de las órdenes necesarias. R:base también permite tener ficheros con muchos más registros que en dBase II, y permite abrir hasta 40 ficheros de datos a la vez. Aunque R:base tenga muchas características de las que carece dBase II, la popularidad de dBase II significa que hay un número mucho mayor de usuarios, distribuidores y consultores de formación para dBase II que para R:base.

Si no va a realizar una gran cantidad de trabajo, su mejor elección es un DBMS que sea fácil de aprender y de usar. Infoscope, de Microstuf, permite introducir y recuperar datos fácilmente. También tiene muchas características nuevas que lo hacen popular. Por ejemplo, Infoscope «apila» ficheros para que usted pueda ver la relación entre sus bases de datos. Cuando usted examina la información que hay en un fichero, una parte de éste se muestra en la pantalla. Si examina un fichero distinto, ese fichero se coloca sobre el anterior; si quiere ver el primer fichero de nuevo, puede «elevarlo» hasta la parte de arriba de la pila. Otras características útiles de Infoscope son el uso del color (en computadoras que tienen monitores de color) para diferenciar las bases de datos, una interfaz de órdenes amistosa, pantallas de ayuda con mucha más información que otros DBMS y, búsqueda y ordenación extremadamente rápidas.

Infoscope también tiene algunos inconvenientes. Por ejemplo, el tamaño de las bases de datos está limitado, y por consiguiente la longitud de los registros. Sin embargo, para el trabajo que realiza, es un programa excelente y también mucho menos caro que la mayor parte de los otros DBMS.

Un excelente DBMS que no requiere programación, *PC-FILE* de Jim Button, es un programa de software compartido. (La contribución sugerida es 35 $). PC-FILE permite construir bases de datos sencillas muy fácilmente. El programa está escrito en BASIC, así que puede modificarlo usted mismo si sabe BASIC. PC-FILE tiene un número de características útiles que le permiten funcionar con otros programas comerciales, aunque la documentación no es muy completa.

Como puede ver, hay un amplio rango de características y opciones disponibles para los DBMS. Esta discusión ha descrito sólo unos cuantos DBMS; hay muhos otros. Entre los programables están *T.I.M.*, de Innovative Software, *DB Master*, de Stoneware, *DataKeep*, de Mathtech, *KnowledgeMan*, de Micro Data Base Systems, y *RL-1*, de ABW. Paquetes como *PFS:File, de Software Publishing Corporation, InfoStar*, de MicroPro, y *Data Design*, de Insoft, no requieren destreza en programación y son a menudo mucho más apropiados para usuarios principiantes.

SOFTWARE INTEGRADO

Un paquete de *software integrado* contiene unos cuantos programas de aplicación que pueden compartir información. Generalmente, el software integrado contiene al menos una hoja de cálculo, un sistema de gestión de base de datos, y operaciones gráficas (tratadas más tarde en este capítulo), pero a menudo contienen otros programas, tales como procesadores de texto, comunicaciones, planificación de proyectos y contabilidad para negocios. La idea principal que está detrás del software integrado es que puesto que cada parte del programa puede usar los datos de todas las demás partes, usted no necesita pasar una gran cantidad de tiempo transfiriendo datos de una parte a otra.

El software integrado ofrece varias ventajas. Se puede pasar fácilmente información entre aplicaciones. Por ejemplo, usted podría hacer una gráfica con las estadísticas obtenidas con la hoja de cálculo. Además, su interacción con cada parte del software es similar, puesto que la misma estructura de órdenes se suele usar en todo el programa. También hay ventajas en el precio, puesto que el software integrado le permite comprar la mayor parte de los programas de aplicación que necesita para su computadora, en un único paquete.

Recuerde que puede comprar las aplicaciones que necesite por separado, pero puede que no consiga transferir información desde los ficheros de un programa a los de otro. Además, aunque el software integrado suele costar más que los paquetes individuales, el costo es menor que el precio combinado de los paquetes separados.

Hay unos cuantos buenos programas integrados en el mercado, pero hay docenas de programas regulares o malos. Antes de comprar software integrado, considere sus necesidades cuidadosamente y busque el software que las satisfaga y sea fácil de usar. Debería hacer las siguientes preguntas cuando examine un software integrado:

- ¿Son buenos los programas individuales? La conveniencia de combinar muchos programas en uno se anula rápidamente si los programas individuales no funcionan bien. Asegúrese de que los programas que más utilizará son adecuados a sus necesidades.
- ¿Es realmente integrado el paquete o es simplemente un grupo de programas vagamente relacionados? Una forma fácil de determinarlo es comprobar si la interacción con cada programa del paquete es realmente similar. Por ejemplo, compruebe si las teclas de función tienen un significado similar en cada programa.
- ¿Se puede realmente pasar información de un programa a otro? Algunos paquetes integrados requieren que usted ejecute programas especiales de utilidad antes de utilizar los datos de una parte del programa en otra parte. Los programas de este tipo son prácticamente lo mismo que usar software no relacionado, porque la mayor parte del software no integrado trae programas que permiten utilizar otro software popular.
- ¿Realmente necesita todos los programas que hay en el paquete? Si usted sólo ejecuta dos tipos de programas, la integración probablemente no le será de utilidad. Sin embargo, si es importante que varios de sus programas utilicen los mismos datos, el software integrado es una buena idea.

Uno de los primeros paquetes de software populares para el MS-DOS fue *1-2-3,* desarrollado por Lotus Development Corporation. 1-2-3 es una combinación de una hoja de cálculo, un sistema de gestión de base de datos y un programa de gráficos de negocios. El producto fue un éxito instantáneo y otras compañías de software se han apresurado a copiarlo, comenzando así la «revolución» del software integrado. Un nuevo producto de Lotus, llamado

Symphony, tiene todas las características del 1-2-3 y también incluye comunicaciones telefónicas y procesamiento de textos.

De los paquetes integrados, 1-2-3 es todavía uno de los mejores. Su manual está bien escrito y es fácil de usar, haciéndolo perfecto para el principiante. 1-2-3 permite realizar complicadas tareas pulsando simplemente una o dos teclas. Además muchas compañías ofrecen cursos para 1-2-3.

Otro paquete integrado excelente es *SuperCalc3,* desarrollado por Sorcim. Este paquete ofrece las mismas ventajas que 1-2-3, pero sus gráficos de negocios son superiores. Si está familiarizado con las versiones más antiguas de SuperCalc, sabrá que la hoja de cálculo de Sorcim es muy buena y tiene una gran cantidad de potencia. El manual de SuperCalc3 no está tan bien escrito como el de Lotus, pero *10 Minutes to SuperCalc* proporciona una introducción excelente para utlizar el programa.

Algunos otros paquetes populares son *Context MBA,* de Context Management Systems, *Framework,* de Ashton-Tate, *InteSoft,* de Schuchardt Software Systems, *The Creator,* from Software Technology for Computers, *IT,* de Martin Marietta, *Number Cruncher,* de Pyramid Data, y *Open Access,* de Software Products International. Hay docenas de programas integrados más.

PROGRAMAS DE CONTABILIDAD

La mayor parte de las pequeñas empresas que compran una computadora MS-DOS esperan usarla para contabilidad. Hay programas de contabilidad para cada tipo de función contable. Desgraciadamente, elegir programas contables es más difícil que elegir otros programas de aplicaciones, puesto que gente de diferentes departamentos se ve implicada frecuentemente en la introducción de la computadora en el proceso contable.

Los programas de contabilidad son como DBMS preprogramados. Las funciones contables corrientes como cuentas a pagar e inventario requieren un conjunto de datos, procedimientos para introducir los datos y un método para hacer informes sobre los datos que forman parte del conjunto de datos. Aunque puede hacer esto con un DBMS, los paquetes contables son más fáciles de usar porque toda la programación ya ha sido realizada.

La mayor parte de los programas de contabilidad se utilizan para introducir datos de negocios y dar informes, tales como resúmenes mensuales y trimestrales, efectivo en mano, etc. Las funciones contables más comunes son el libro mayor, cuentas corrientes, e inventario. Cada programa tiene una forma ligeramente diferente de realizar cada una de estas tareas. Los programas contables varían tanto en la forma en que toman sus datos de negocios como en el tipo de informe que producen.

Los paquetes contables pueden ser difíciles de utilizar al principio. Puesto que no los escribió usted mismo, no pueden ajustarse exactamente a sus necesidades. Esto puede que no suene como un obstáculo serio, puesto que con un procesador de texto, usted usualmente se acostumbra a tales diferencias y hace

modificaciones en su estilo de escritura. Sin embargo, en un paquete contable, las diferencias entre lo que el programa hace con los datos y cómo los organiza usted usualmente, pueden hacer que sea extremadamente difícil acostumbrarse. Casi todo el mundo estará de acuerdo con que la mejor palabra para describir la primera semana (o el primer año) con un nuevo programa de contabilidad es «frustrante». Pero a la larga, sus libros estarán más organizados y serán más fáciles de usar.

Los paquetes contables suelen ser más caros que otros tipos de software, costando frecuentemente entre 500 $ y 5.000 $, según el número de tareas contables que realicen. Afortunadamente, las tiendas de computadoras suelen dejar que los pruebe antes de comprarlos. Esto es bueno, puesto que cada programa maneja los datos de una forma diferente. Antes de comprar el programa, debería estar seguro de que usted (o la persona que más utilizara el programa) está cómodo con el tipo de interacción requerido. Muchos programas le permiten que adapte a sus necesidades la forma de introducir la información en el programa, así como la forma en que el programa visualiza los resultados.

Si tiene un contable o tenedor de libros, pida su opinión antes de comprar un programa. Muchas pequeñas compañías pasan frecuentemente su sistema contable a una computadora sin consultar a la gente que más trabaja con las cifras. El coste de incluir a esta gente en el proceso de decisión se justifica fácilmente si el resultado final es pasar menos tiempo con los libros. Los contables normalmente pueden decir si un paquete contable particular se ajusta a las necesidades de la compañía. En realidad, suele ser útil llevar una muestra de sus datos e informes cuando va a comprar software de contabilidad.

Algunos contables conocen a otras personas que usan computadoras MS-DOS y que puede que conozcan algunos buenos programas. También puede preguntar a alguien que haya probado varios paquetes, cuál elegir y por qué. Las revistas de computadoras, tales como las descritas en el capítulo 7, suelen tener buenas revisiones de paquetes de contabilidad que frecuentemente han sido escritas por contables.

Aunque hay una gran variedad de paquetes, puede que verdaderamente usted tenga necesidades especiales que ningún programa pueda satisfacer. En este caso, lo mejor que puede hacer es pagar a un consultor para que le escriba un programa que se ajuste a sus necesidades. Esto no es tan raro ni tan caro como parece. En realidad, puede que el consultor ya haya escrito un programa similar que puede modificar ligeramente para usted. Tenga en cuenta, sin embargo, que un programa contable, escrito por un consultor, puede costar hasta tres o cuatro veces lo que un paquete de programas. Muchas compañías pagan a un consultor, que está familiarizado con muchos de los paquetes a la venta, para que les ayude a decidir cuál comprar.

Al hacer frente a lo que parece ser un montón de paquetes idénticos, puede ser tentador elegir simplemente uno, realizado por un fabricante muy conocido. Resista la tentación. Las decisiones apresuradas suelen ser una

mala estrategia y esto es verdad, especialmente con los paquetes contables. Una vez que usted meta todos sus registros en un sistema, es muy caro cambiarlos a otro.

Algunas características que debería buscar cuando compre un paquete contable son

- Pantallas de entrada adaptables.
- Informes adaptables.
- Interfases con los sistemas populares de gestión de base de datos.
- Disponibilidad y coste de las nuevas versiones de los programas.
- Seminarios dados por el fabricante para los contables.
- Cantidad de asistencia que el distribuidor obtiene del fabricante.
- Recomendaciones de empresas de contabilidad importantes.

The Structured Systems Group es bien conocido por tener una de las primeras y más fáciles series de paquetes, con una asistencia al cliente excelente. Las series *EasyBusiness* de Information Unlimited Software cubren un rango muy amplio de aplicaciones. *Solomon* de Computech Group, es uno de los paquetes favoritos para contables, porque los usuarios pueden especificar el formato de los informes que produce. Este programa es también fácil de aprender. Las series *RealWorld,* de MicroBusiness Software y *Desktop Accountant,* de Rocky Mountain Software Systems, son también muy populares.

PROGRAMAS DE PLANIFICACION DE PROYECTOS

Las computadoras están muy bien para organizar listas, especialmente listas largas. Si su trabajo implica planificación de proyectos, usted sabe lo difícil que es organizar tareas individuales en el orden en que serán realizadas. Afortunadamente, hay muy buen software MS-DOS para hacer esto.

El mayor quebradero de cabeza en la planificación de proyectos es el establecer un orden de precedencia, que determina si se necesita que otras subtareas estén hechas antes de la subtarea actual, o si esas tareas dependen de que la tarea actual se complete primero. Por ejemplo, si está construyendo una casa, hay que construir la estructura antes de instalar los muros; así, los instaladores de muros no hacen falta hasta que hayan terminado los constructores de la estructura.

Si cada trabajo se realizara haciendo sólo una tarea cada vez, usted simplemente tendría una lista unidimensional de tareas con cada tarea ejecutada después de otra. Sin embargo, muchas tareas preparatorias para distintos trabajos pueden realizarse al mismo tiempo. En el ejemplo de la casa, aunque los instaladores de los muros no necesiten presentarse antes de que los construc-

tores de la estructura hayan terminado, los materiales para los muros pueden ser entregados y cortados a su tamaño, mientras se construye la estructura. De esta manera, es importante realizar tareas no relacionadas, al mismo tiempo. La figura 6-6 muestra un ejemplo del tipo de salida que podría producir un programa de planificación de proyectos.

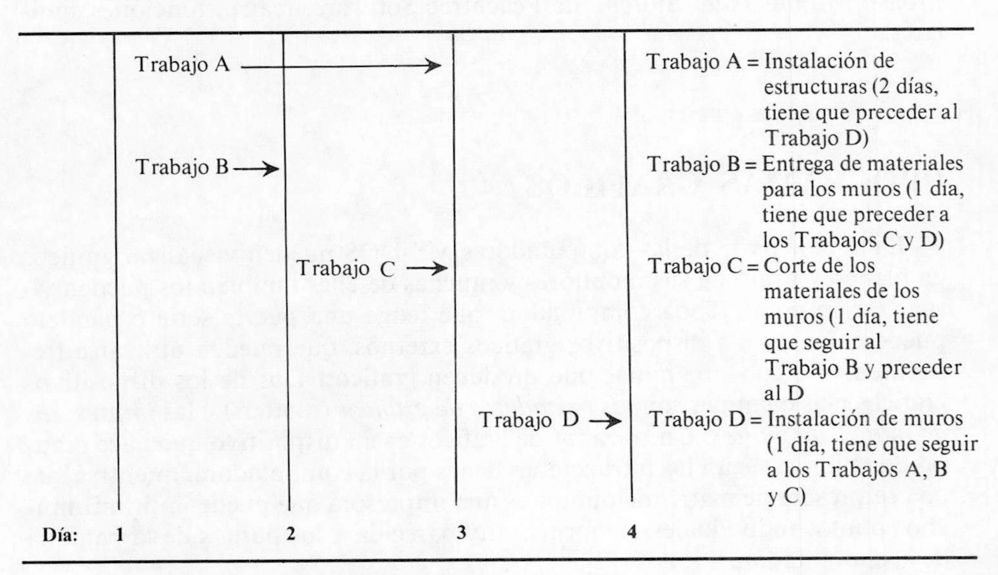


Figura 6-6. Ejemplo de salida de un paquete de software de planificación de proyectos

El software de planificación de proyectos le ayuda a organizar una innumerable cantidad de subtareas para que pueda planificar un trabajo grande. Usted describe cada subtarea del proyecto y lista todas las otras subtareas que necesitan estar terminadas antes de empezar dicha subtarea. Una vez que se le ha dado al programa toda la entrada, éste le muestra cómo organizar el proyecto, indicando qué tareas pueden ser realizadas simultáneamente y el tiempo transcurrido en el trabajo total.

Una enorme ventaja del software de planificación de proyectos es la capacidad de realizar cálculos del tipo «qué-pasa-si». Por ejemplo, puede fácilmente proyectar qué pasaría si un cierto trabajo no se terminase a tiempo. Se pueden corregir números y revisar las predicciones, mientras un proyecto progrese en la vida real.

Muchos paquetes de planificación de proyectos le ayudan incluso a estimar los costes de trabajo. Por ejemplo, si introduce el coste de las horas de un trabajador para cada tipo de trabajo, puede calcular los costes totales para la duración del trabajo. Si los materiales varían de precio según cuando se adquieran puede incluir este factor en sus cálculos.

Los dos paquetes de planificación para el MS-DOS más populares, *Micro-GANTT,* de Earth Data y *Project Scheduler,* de Scitor, usan el método Gantt, una popular herramienta de gestión para planificación. Project Scheduler tiene muchas características interesantes, como tablas internas de días festivos y la capacidad de leer ficheros de VisiCalc y SuperCalc. MicroGANTT permite hacer ficheros de salida que pueden ser leídos por paquetes de hoja de cálculo. *Project Management System,* de Peachtree Software, realiza funciones similares.

PROGRAMAS GRAFICOS

La mayor parte de las computadoras MS-DOS pueden visualizar gráficos en blanco y negro en sus monitores y muchas de ellas también los pueden visualizar en color. Toda computadora que tenga una puerta serie o paralelo puede conectarse a dispositivos gráficos externos, que pueden utilizarse frecuentemente con programas que producen gráficos. Dos de los dispositivos gráficos más comunes son los *trazadores de gráficos* (plotters) y las *impresoras de matriz de puntos.* Un trazador de gráficos es un dispositivo que hace dibujos con pluma, según las intrucciones dadas por la computadora, mientras que una impresora de matriz de puntos es una impresora que puede imprimir muchos puntos individuales, de forma muy parecida a los puntos de la pantalla de la computadora.

Gráficos de negocios

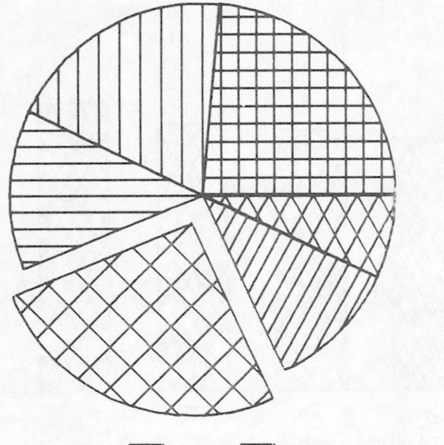
El software para gráficos de negocios visualiza datos numéricos en histogramas, diagramas de sectores y diagramas de líneas. El dicho de que una imagen vale más que mil palabras se utiliza a menudo para describir los gráficos de negocios. Sin embargo, los programas para gráficos de negocios son más útiles como método para resumir largos informes, que como una forma de presentar información desde otro punto de vista. El ser capaz de reducir un conjunto de números a un diagrama de sectores puede que le permita ver la relación entre los números, pero puede también oscurecer su magnitud.

Hay muchos factores a considerar al elegir un paquete de gráficos para negocios. La cuestión más importante es si puede producir gráficos en su hardware. Por ejemplo, muchos paquetes sólo funcionan si usted tiene un monitor (casi todos los que se ejecutan en el IBM PC requieren un adaptador de color y un monitor); otros sólo funcionan con impresoras gráficas (tales como una Epson) o con trazadores de gráficos. Algunas computadoras y algunos adaptadores gráficos de color añadidos, ofrecen una resolución más alta que la tarjeta adaptadora de color estándar de IBM.

Otra consideración importante al elegir un paquete es si es fácil introducir los datos. Muchos paquetes pueden leer los datos producidos por otros programas de aplicación, como hojas de cálculo y DBMS. Otros requieren que los introduzca manualmente. Aunque suele ser fácil descifrar cómo obtener los datos para el programa gráfico, puede llevar tiempo el hacerlo así.

Si quiere que otras personas vean los gráficos que usted produce, asegúrese de conseguir un paquete que pueda funcionar tanto con una impresora como con un trazador de gráficos. Aún con los mejores gráficos, las impresoras son lentas y usualmente no producen buenos gráficos de negocios. Muchos fabricantes venden trazadores de gráficos de bajo costo que producen mucho mejores resultados. La mayor parte de los trazadores de gráficos pueden producir salida multicolor, tanto en papel como en transparencias. No necesita tener pantalla gráfica para usar una impresora gráfica o un trazador de gráficos. En realidad, puede comprar un trazador de gráficos de dos plumas de color por menos que un monitor color y su adaptador.

Acme Scientific Sales, 1983



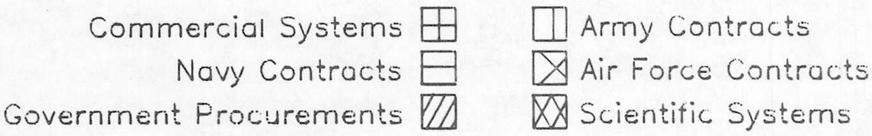


Figura 6-7. Un ejemplo de diagrama de sectores

Antes de comprar un paquete de gráficos de negocios, compruebe que el paquete ofrece suficientes tipos de diagramas diferentes, para sus diferentes tipos de datos. Los diagramas de sectores (mostrados en la figura 6-7) y los histogramas (mostrados en la figura 6-8) sólo son útiles para mostrar datos unidimensionales, tales como las ventas de los distintos departamentos en un año. Los datos unidimensionales son bastante comunes, pero usted no suele necesitar gráficos para ver la relación entre los números. La mayor parte de los paquetes gráficos dibujan uno de estos tipos de diagramas o ambos.

Los datos bidimensionales se muestran con diagramas de líneas, histogramas apilados o agrupados o con histogramas tridimensionales. Aquí es donde los gráficos de negocios se hacen más útiles; no es tan fácil ver las diferencias cuando se están comparando datos con diferentes relaciones al mismo tiempo. Los datos bidimensionales muestran la relación entre elementos según dos criterios diferentes. Por ejemplo, la comparación de las ventas de diferentes

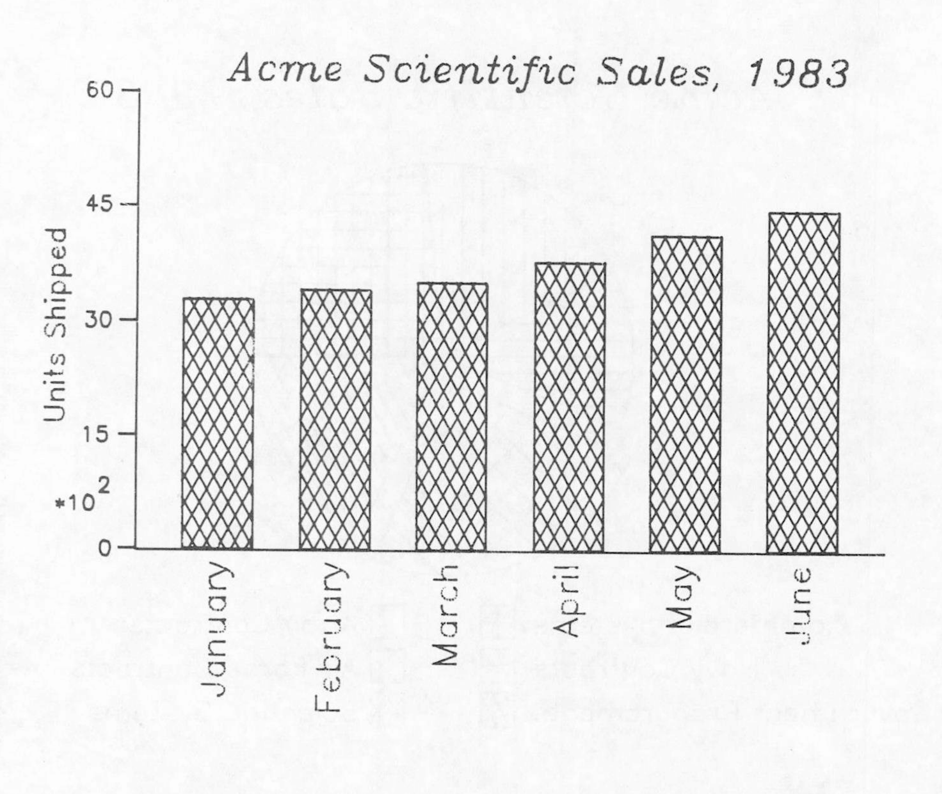


Figura 6-8. Un ejemplo de histograma

departamentos a lo largo de varios años utiliza datos bidimensionales: la primera dimensión corresponde a los diferentes departamentos y la segunda a los diferentes años. Estos diagramas permiten comparar fácilmente los departamentos entre sí en un determinado año y comparar un departamento con sí mismo a lo largo de varios años.

Las figuras 6-9, 6-10, 6-11 y 6-12 muestran respectivamente un diagrama de líneas, un histograma apilado, un histograma agrupado y un histograma tridimensional. Aunque parezcan completamente distintos, todos permiten comparar dos tipos de datos a la vez.

Un paquete gráfico barato y fácil de usar es *PCcrayon,* de PCsoftware. Produce diagramas de datos unidimensionales y bidimensionales y puede hacer una variedad de diagramas en la pantalla de color. También puede copiar los contenidos de la pantalla en una impresora de matriz de puntos. No obstante, los gráficos de negocios son sólo una parte del paquete; también puede usar el

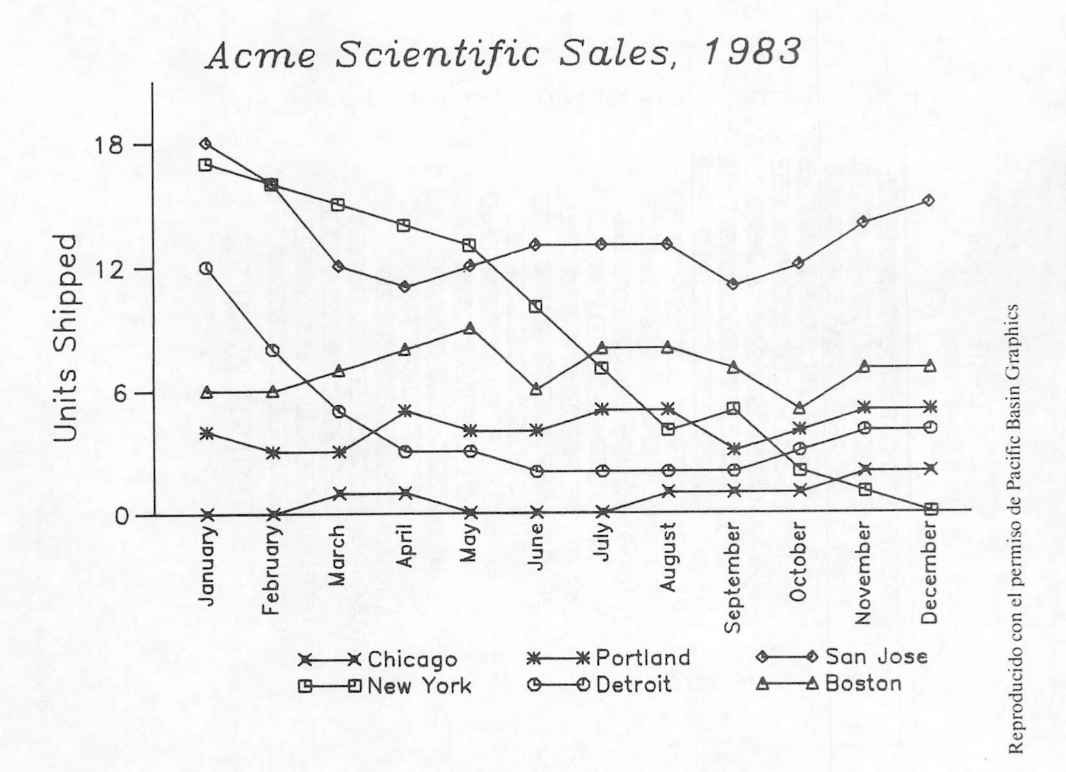


Figura 6-9. Un ejemplo de diagrama de líneas

programa para pintar, utilizando colores y texto. Es un paquete versátil, aunque el manual no explica todas sus características.

PBG 100, de Pacific Basin Graphics, produce todo tipo de diagramas en monitores, impresoras y trazadores de gráficos. El software se ejecuta en CP/M y UNIX además de en MS-DOS, así que una compañía que utilice distintos sistemas operativos puede usar el software en todas sus computadoras. Todas las figuras de esta sección fueron producidas con PBG 100 y un trazador de gráficos.

GSS-Chart, de Graphic Software Systems, es un paquete más sofisticado. Puede visualizar diagramas en dispositivos de muchos fabricantes distintos, puesto que GSS suministra software gráfico a muchas compañías de hardware. La interfaz de usuario para GSS-Chart puede que proporcione demasiada ayuda para las personas familiarizadas con el programa, pero la salida es muy buena.

EnerGraphics, de EnerTronics, es otro programa de alta calidad que per-

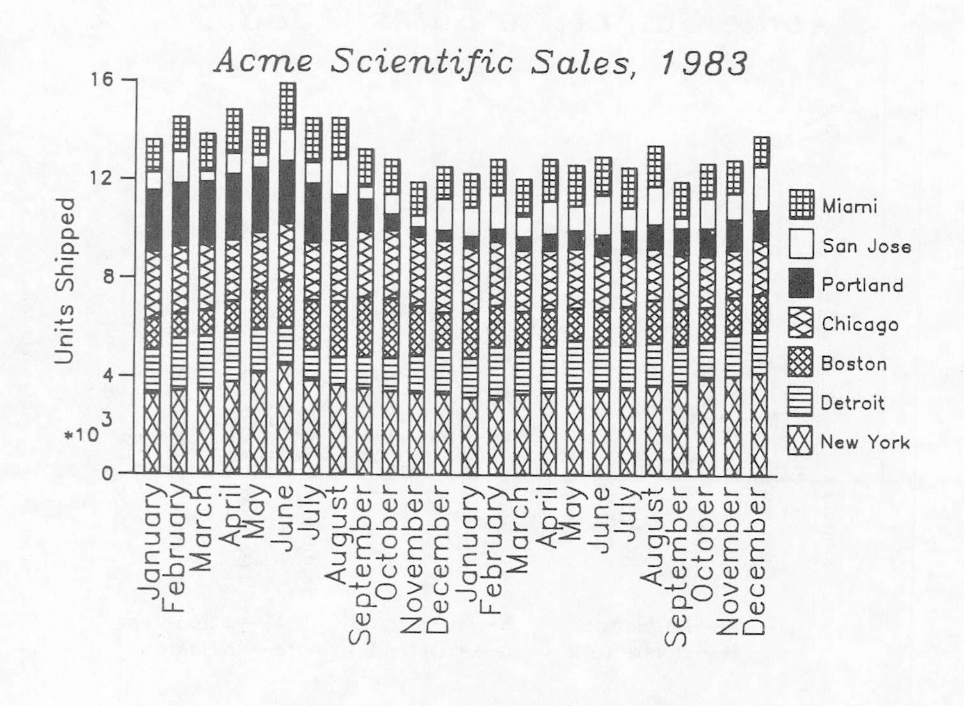


Figura 6-10. Histograma apilado

mite visualizar gráficos en la pantalla o imprimirlos en una impresora gráfica. Puede usar también EnerGraphics para crear dibujos sofisticados de dos y tres dimensiones (como representaciones de ingeniería) y otras funciones. Otro buen paquete es *Grafox,* de Fox & Geller. Hay también docenas de programas de dominio público que realizan gráficos de negocios.

Dibujo de imágenes

Naturalmente, se pueden hacer muchas otras cosas con los programas gráficos, además de los gráficos de negocios. Hay cierto número de paquetes gráficos que permiten dibujar imágenes en vez de visualizar gráficamente los datos. Igual que un procesador de texto, una vez que usted ha dibujado una imagen, usa el programa gráfico para editarla hasta que cumpla las especificaciones exactas. Normalmente puede producir «proyecciones», que son con-

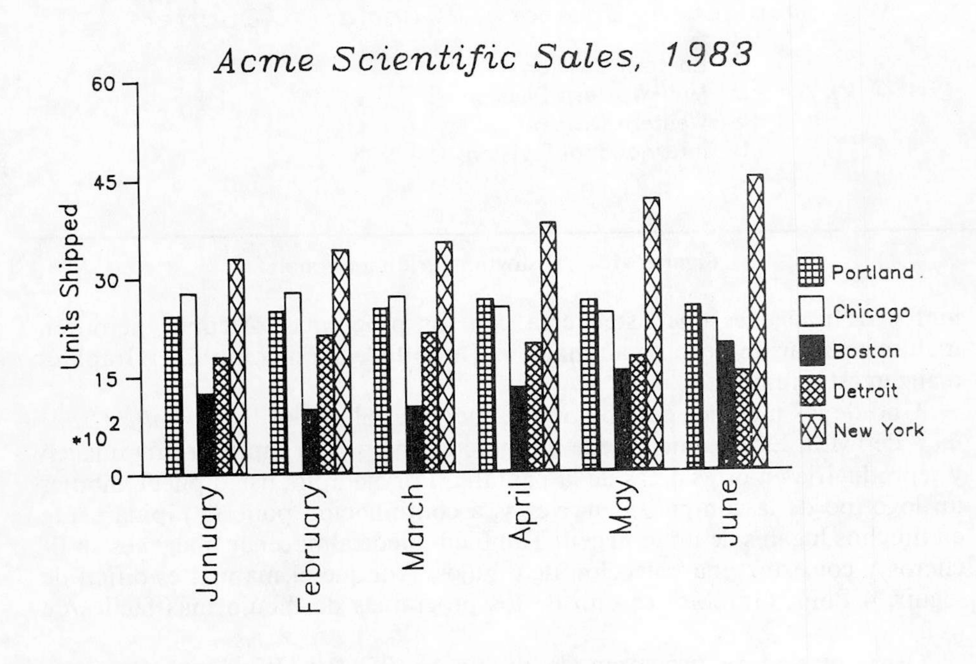
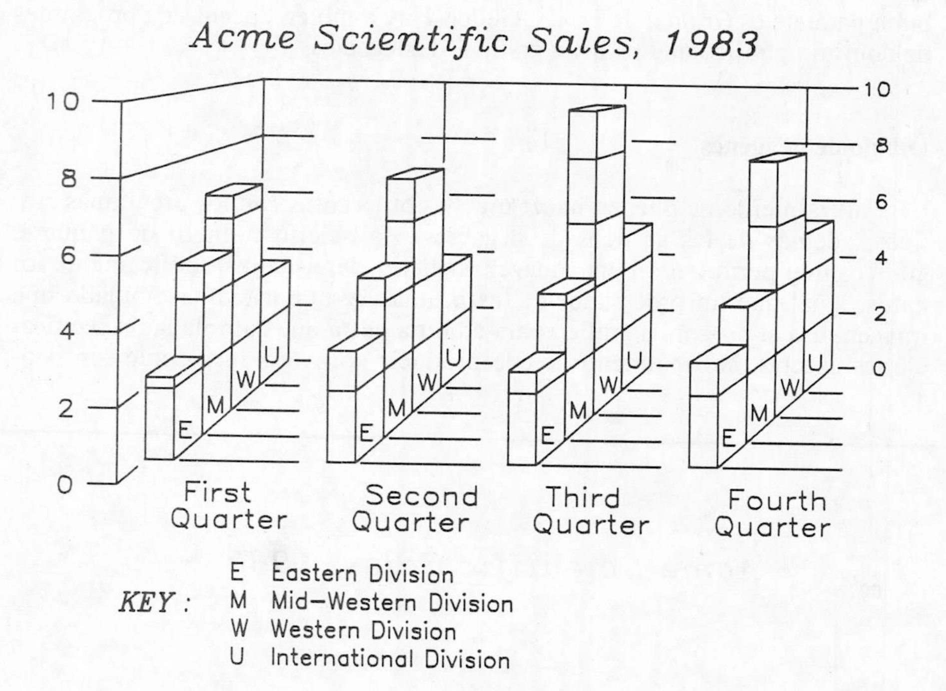


Figura 6-11. Histograma agrupado



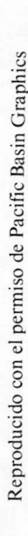


Figura 6-12. Histograma tridimensional

juntos de imágenes que usted crea con los programas. Algunos permiten incluso realizar un poco de animación. Tanto PCcrayon como EnerGraphics realizan estas funciones.

Uno de los métodos de dibujo más nuevos es el que utiliza *4-Point Graphics,* de IMSI. Este paquete tiene un método único para capturar una imagen y reproducirla en otras áreas de la pantalla. Por ejemplo, usted puede dibujar un logotipo de la compañía una vez y, a continuación ponerlo rápidamente en muchos lugares de una imagen. También puede almacenar imágenes en ficheros y construir una colección de dibujos. Aunque el manual es difícil de seguir, 4-Point Graphics es uno de los programas de dibujo más fáciles de usar.

Otro interesante programa de dibujo es *PC-DRAW,* de Micrografix. PC-DRAW utiliza símbolos para construir diagramas de flujo, diagramas eléctricos y otros dibujos que utilizan símbolos. Aunque crear símbolos necesita alguna paciencia, una vez que tenga sus símbolos, crear dibujos con ellos es muy fácil.

En el extremo más caro del mercado de los gráficos, *EASEL*, de Time Arts, Inc., es un producto maravilloso. Está diseñado para artistas gráficos y se puede utilizar para reproducción orientada a pantalla o video-reproducción. La mayor parte de la entrada se hace con un lápiz digitalizador (un dispositivo especial de entrada para dibujar en una tableta gráfica y transferir el dibujo instantáneamente a la pantalla). EASEL permite usar hardware especial para conseguir dibujos en color de muy alta resolución, con muchas computadoras MS-DOS.

Otros productos para dibujo son *Mantis*, de Suttle Enterprises, *Ultradraw*, de Ultragraphics Systems, *Lenipen*, de Duncan-Atwell Computerized Technologies (que permite usar un lápiz óptico para dibujar imágenes), y *TELI-graph*, de Microtaure.

Diseño ayudado por computadora

Si es usted un arquitecto o está interesado en utilizar su computadora para diseño o delineación, hay un cierto número de productos que puede usar. Estos paquetes proporcionan características tan sofisticadas como el dibujo en tres dimensiones y están escritos especialmente para las necesidades de los diseñadores. Sin embargo, a menudo cuestan de cinco a diez veces más que los paquetes gráficos que acabamos de describir. Naturalmente, ninguno de estos paquetes son tan sofisticados como los sistemas profesionales que cuestan alrededor de 50.000 $. Sin embargo, ofrecen un medio menos caro de empezar a diseñar con la computadora.

AutoCAD, de Autodesk, Inc., es un paquete excelente que permite editar los dibujos con un ratón. Se pueden crear espacios bidimensionales y construir planos, así como otros dibujos (tales como diagramas de organización). *MicroCAD*, de Computer Aided Design, es similar pero proporciona mucha más flexibilidad, puesto que se pueden dibujar planos en tres dimensiones. El dibujo en perspectiva de MicroCAD es una gran ayuda para los arquitectos cuando diseñan exteriores, porque le permite ver los planos de un edificio desde todos los ángulos. Otros paquetes de diseño son *Drawing Processor*, de BG Graphics, y *CADPLAN*, de Personal CAD Systems.

SOFTWARE DE COMUNICACIONES

Un *modem* es un dispositivo hardware muy común que se añade a las microcomputadoras. Un modem permite comunicar con otras computadoras a través de líneas telefónicas normales. Algunas computadoras traen modems como característica estándar. Los modems permiten usar un amplio número de servicios de información (como grandes bases de datos y asociaciones de usuarios) que están disponibles en las computadoras de otras personas.

¿Cómo funciona un modem? El proceso es en realidad muy simple. Con

modems, las líneas de la compañía telefónica se convierten en un cable muy largo entre computadoras. De la misma forma que la computadora envía caracteres a la impresora a través de un cable, un modem envía caracteres a través de la línea telefónica a otro modem o a otra computadora. Desgraciadamente, la mayor parte de los modems no permiten comunicar muy rápidamente. Lo más frecuente es que transmitan 30 caracteres por segundo (300 baudios) ó 120 caracteres por segundo (1.200 baudios).

Se necesita algo más que un modem para intercambiar datos con otras computadoras; también se necesita software de comunicaciones para decirle al modem cómo enviar la información. A continuación, se citan las tareas que el software de comunicaciones puede ejecutar:

- En su forma simple, el software de comunicaciones conecta su terminal directamente a la otra computadora. Esto se llama *emulación de terminal.* Por ejemplo, si usted se conecta con una computadora de gran tamaño, el software de comunicaciones enviará cada carácter que usted escriba a la otra computadora e imprimirá en su pantalla todos los caracteres que escriba la otra computadora. Hay muchas compañías que le permitirán el acceso a la información que hay en sus computadoras de gran tamaño, si usted tiene un programa de comunicaciones y un modem.
- El software de comunicaciones también permite enviar ficheros texto de una computadora a otra. Por ejemplo, se puede enviar un fichero de informes, por línea telefónica, a alguien que tenga una computadora y un modem. Enviar un fichero texto con el programa de comunicaciones es más rápido y más exacto que volver a escribir el fichero y enviarlo.
- Algunos programas de comunicaciones permiten incluso enviar programas por línea telefónica. Para poder hacerlo, las dos personas necesitan, o bien ejecutar los mismos programas de comunicaciones o tener dos programas que usen el mismo método para transmitir y recibir ficheros de programas. Puesto que los programas son como ficheros con caracteres especiales, la mayor parte de los programas de comunicaciones que permiten enviar y recibir ficheros texto, también permiten enviar y recibir programas.
- Algunos modems son capaces de marcar el número de teléfono (esta característica se denomina *auto-marcado*), pero necesitan software que proporcione las órdenes adecuadas. Muchos programas de comunicaciones permiten guardar números de teléfonos internamente para que el programa los marque. Esto hace aún más fácil la utilización del modem: usted simplemente le dice a la computadora que llame a otra determinada computadora.

Casi todos los programas de comunicaciones pueden realizar la emulación del terminal y la transferencia de ficheros texto, y la mayor parte pueden realizar transferencia de programas y auto-marcado.

Las otras características que debería buscar en el software de comunicaciones son la facilidad de uso (la mayoría son muy sencillos) y la compatibilidad con otros programas. Desgraciadamente, esto último es difícil de determinar. Debería conocer el *protocolo de comunicaciones* (el método que las computadoras utilizan para asegurarse de que se envían los datos correctos) que utilizan las computadoras a las que usted llama, porque el software de comunicaciones tiene que usar el mismo protocolo. No hace falta que sepa lo que hacen los protocolos, sólo necesita el nombre de dichos protocolos. Dos de los protocolos más comunes son XON/XOFF, también conocido como «CONTROL-Q/CONTROL-S, y MODEM7, un protocolo que se usa casi exclusivamente por microcomputadoras. Los paquetes de comunicaciones suelen costar 50 $ y 200 $.

Algunos paquetes de comunicaciones a la venta tienen características suficientes, tanto para principiantes como para usuarios avanzados. *PC-TALK,* de Headlands Peess, es un paquete de comunicaciones excelente. Es muy fácil de usar, aunque el manual no está claramente escrito. PC-TALK puede comunicar con los protocolos XON/XOFF y MODEM7, y puede almacenar números y marcar, utilizando un directorio de números de teléfono. Igual que PC-Write, PC-TALK es software compartido, lo que significa que usted puede obtener una copia gratuitamente. (La contribución solicitada es de 35 $ si decide que quiere usarlo). El disquete de distribución contiene el código fuente en BASIC. Así, si sabe programar en BASIC, puede modificar el programa.

Otro producto muy bueno es *Crosstalk XVI,* de Microstuf. Crosstalk ha sido uno de los programas de comunicaciones mejor vendidos durante años, tanto en CP/M como en MS-DOS, a causa de sus posibilidades y facilidad de uso. El manual es excelente y el programa también proporciona ayuda por pantalla. Una característica especialmente útil es su capacidad para almacenar órdenes en un fichero y, a continuación decirle a Crosstalk que use dicho fichero (esto es similar a los ficheros batch del MS-DOS). Microstuf también vende *Transporter,* que permite enviar y recibir ficheros mientras usted está lejos de la computadora (como por ejemplo, por la noche, cuando las tarifas telefónicas son bajas).

Otros programas populares son *Asynchronous Communications Support,* de IBM, *Micro Link II,* de Digital Marketing, *LYNC,* de International Software Alliance, *ReadiTerm,* de ReadiWare, y *Ascom,* de Dynamic Microprocessor Associates.

APRENDIZAJE ASISTIDO POR COMPUTADORA

A la mayor parte de la gente le gusta aprender leyendo libros u oyendo a un conferenciante; esto se debe probablemente a la forma en que les enseñaron en la escuela. Los libros y el material impreso son aún los medios más comunes de aprender acerca de las computadoras (después de todo, usted está leyendo un libro en este momento). Sin embargo, las computadoras son tam-

bién un medio valioso para el aprendizaje. Muchas compañías han experimentado con el aprendizaje asistido por computadora, durante varios años, y el software educativo suele ser bueno.,

Las computadoras se pueden utilizar para enseñar temas que no tienen nada que ver con ellas mismas. En realidad, muchas compañías de formación están usando enseñanza asistida por computadora en una amplia variedad de campos, desde reparación de automóviles hasta cursos para enfermeras. Los programas tratados en la sección que está a continuación, enseñan exclusivamente MS-DOS; la sección siguiente se dedica a otros programas de enseñanza.

Aprendizaje de computadoras

Los programas proporcionan un método ideal para aprender lo relativo a las computadoras y al software, porque son *interactivos:* usted introduce información y ellos responden. En el nivel más simple, esta interacción es lo equivalente a pasar la página. Cuando se ha terminado de leer el texto de la pantalla, se pulsa una tecla para visualizar más información. Una interacción más avanzada tiene lugar cuando la computadora hace una pregunta y, a continuación realiza una acción basada en la respuesta. Por ejemplo, si la computadora le hace una pregunta de comprobación y usted se equivoca, puede repetir la información que usted no había aprendido.

Sin embargo, hay muchos inconvenientes para aprender con computadoras. Estas son relativamente ineficientes. La mayor parte de los programas que enseñan MS-DOS sólo lo tratan superficialmente y aún así pueden necesitar 5 disquetes para su almacenamiento. Otro problema es que no se pueden usar estos programas como una guía de referencia fácil, como un libro o una revista. Puede que usted esté limitado a un avance hacia adelante a través del disco y que no pueda retroceder para volver a comprobar un determinado punto. Además muchos programas permiten poca o ninguna experimentación con las órdenes que no tratan.

Un buen programa de formación tiene que mantener la atención de los usuarios, asegurarse de que entienden cada punto antes de avanzar al siguiente y permitirles probar ejemplos de lo que están aprendiendo. De otra manera, no habría ventajas sobre el libro tradicional o la educación en una clase.

Puesto que la enseñanza de computadoras es muy popular, están a la venta muchos programas tutoriales sobre computadoras. Estos tutoriales cubren muchos temas, desde cómo usar sistemas operativos (como el MS-DOS) hasta la enseñanza de la programación. Varían en calidad y precio y usted debería indudablemente probarlos antes de comprarlos.

PC Tutor, de Comprehensive Software, es un buen ejemplo de un programa bien escrito y barato que presenta una vista general del MS-DOS con abundante interacción. Otro buen paquete es *How To Use Your IBM PC,* de Cdex Corporation. Este programa le lleva lentamente a través de un rango li-

mitado de temas. *Teach Yourself MS-DOS,* de ATI, es mucho menos interesante a causa de su limitada interacción y mal uso de la pantalla.

Programas de aprendizaje creativo

Los programas tutoriales tratados en la sección anterior parecen libros de texto con un poco de interacción, algunos gráficos ostentosos en la pantalla y sonidos ocasionalmente. Hay otros programas educativos para adultos que utilizan métodos de enseñanza completamente diferentes. Aunque pueden parecer juegos, su propósito es aprender, no ganar. Estos programas parecen juegos de estrategia porque tienen un objetivo que usted alcanza utilizando su habilidad mental.

Knoware, de la compañía del mismo nombre, es un paquete que enseña a los hombres de negocios lo que su computadora puede hacer por ellos. El juego introduce algunas funciones corrientes de la computadora (como programación, hojas de cálculo, toma de decisiones financieras y DBMS) mientras le propone un objetivo razonable: convertirse en jefe del consejo de administración y hacerse millonario. Para obtener más dinero y puestos de trabajo más altos, usted tiene que aprender los temas destacados y realizar tareas. El programa recalca el aprendizaje aplicado más que la memorización por repetición. Las tareas son interesantes y usted puede pedir ayuda en cualquier momento.

Millionaire, de Blue Chip Software, es un juego que es también una simulación muy realista de la bolsa. La idea que hay detrás del juego es enseñarle a trabajar con acciones y obligaciones y a aceptar una pérdida elegantemente sin ponerse furioso. Partiendo de que las acciones en un grupo industrial se suelen mover juntas, usted compra, vende y pone opciones mientras el programa lanza noticias financieras. Usted puede incluso pedir dinero prestado para comprar acciones, si se atreve.

Mucha gente está frustrada con su computadora, no porque no pueda hacer lo suficiente, sino porque no pueden escribir suficientemente bien para darle las órdenes. *MasterType,* de Lightning Software, es un método divertido para aprender mecanografía. Usa buenas técnicas educativas y funciona en monitores de color y monocromos.

El *FriendlyWare PC Introductory Set,* de FriendlySoft, es una buena introducción a las computadoras para principiantes. Incluye un manual bien escrito, juegos de aprendizaje, juegos divertidos y una descripción muy buena de cómo se escribe un programa BASIC.

Hay un cierto número de juegos educativos para niños. Sin embargo, a menudo estos programas no son ni educativos ni divertidos. Una excepción a esta regla es *In Search of The Most Amazing Thing,* de Spinnaker Software. Este juego enseña a los niños cómo tomar notas, leer un mapa, negociar y obtener ayuda al tomar decisiones. El juego es mucho más complejo que otros, aunque cada parte es suficientemente simple para la mayoría de los niños de diez años. Spinnaker también hace *Delta Drawing,* un paquete innovador que

enseña geometría y programación. Es similar al lenguaje Logo, pero más fácil y más divertido.

PROGRAMAS QUE MEJORAN EL MS-DOS

Muchos productos pueden hacer que la utilización del MS-DOS sea más fácil o más eficiente. Estos productos se llaman *cápsulas* (shells), porque rodean al MS-DOS con una «cápsula protectora». En vez de darle órdenes al MS-DOS, usted da órdenes a estos productos y ellos interpretan las órdenes para el MS-DOS.

Hay básicamente dos tipos de cápsulas: programas que hacen más fácil dar órdenes al MS-DOS y programas que permiten que el MS-DOS realice más de un trabajo, haciéndolo así multitarea o multiusuario. El primer tipo de cápsula elimina la necesidad de recordar todas las sintaxis de todas las posibles órdenes del MS-DOS (aunque usando el capítulo 3 de este libro, esto le resultaría más fácil). El segundo tipo, las cápsulas multitrabajo, intentan hacer más flexible el MS-DOS. Sin embargo, muchos programas de aplicación no funcionan con ellas.

Programas que ocultan la petición de orden A >

El objetivo de algunos programas cápsula es evitar la introducción directa de las órdenes del MS-DOS. Usted no necesita recordar lo que debe hacer cuando vea la petición de orden A >, porque raramente la ve. Estos programas cambian la forma en que usted interactúa con el MS-DOS, apoyándose en la suposición de que usted no quiere recordar las órdenes MS-DOS. Esto probablemente es verdad; nunca es conveniente tener que mirar la sintaxis de una orden o buscar en un libro la forma de hacer algo. Sin embargo, como vio en los primeros capítulos, hay relativamente pocas órdenes que usted use regularmente, y aprender dichas órdenes es bastante fácil.

Los programas cápsula colocan al usuario en desventaja, porque ningún programa puede predecir todas las necesidades de alguien. Un programa cápsula puede ser suficiente para la mayoría de sus necesidades, pero hay situaciones donde usted realmente necesita dar una orden MS-DOS y un programa cápsula puede estorbarle.

El software más sofisticado de este tipo son programas que usan una parte de la pantalla para ayudarle a elegir opciones. Por ejemplo, *visuALL,* de Trillian, usa el concepto de *ventana temporal* para hacerle más fácil la interacción con el sistema operativo y con sus programas. Con visuALL, se usa un ratón o las teclas de función para moverse arriba y abajo de un menú de opciones (tales como las órdenes del MS-DOS). Si quiere trabajar con ficheros, visuALL le permite seleccionar los ficheros con el mismo método. Esto elimina la mayor parte de la memorización implicada tanto en las órdenes MS-DOS como en los programas de aplicación.

Otro excelente programa es *DESQ,* de Quarterdeck Software. DESQ es similar a visuALL en el uso de la pantalla; sin embargo, DESQ permite ver varios programas diferentes en la pantalla al mismo tiempo. Se puede incluso mover información de una pantalla a otra. Otro programa cápsula menos sofisticado es *1 Dir,* de Bourbaki.

POWER, de Computing!, no hace más fácil dar órdenes, pero simplifica la tarea de manejar directorios. Por ejemplo, cuando se da la orden ERASE, POWER muestra una lista ordenada de todos los ficheros del disco y pone un número delante de cada uno. A continuación, se pueden especificar los ficheros que se quieren borrar, dando los números en vez de los nombres. Esto es especialmente útil para algunas órdenes del MS-DOS que requieren nombres de ficheros.

Programas multiusuario y multitarea

Hasta que Microsoft saque otra versión del MS-DOS que sea multitarea o multiusuario (o ambas cosas), probablemente se puede afirmar que los productos de mejora, que no son MS-DOS, no funcionarán con una gran fiabilidad. Hay muchos de estos programas, pero ninguno funciona con todo el software MS-DOS. El mayor obstáculo para la efectividad de estos programas es el software de aplicación que utiliza características del MS-DOS que no son estándar. Un ejemplo de esto son los programas escritos específicamente para el IBM PC. Tales programas puede que no funcionen en otras computadoras MS-DOS.

Otro problema general con estos programas es que algunas computadoras no son suficientemente rápidas para hacer práctico el software multiusuario o multitarea. Por ejemplo, debido a que el IBM PC tiene una CPU bastante lenta, poner muchos usuarios en ella haría que el sistema funcionase muy lentamente. (Esto no es verdad en computadoras MS-DOS con CPU muy rápida, como Radio Shack Model 2000).

El mercado para el software MS-DOS está cambiando a causa de las mejoras del MS-DOS y a la búsqueda de los fabricantes de hardware para encontrar la posición perfecta en el mercado del MS-DOS. No se mencionan programas en esta sección, puesto que la viabilidad de alguno de estos productos será cambiada radicalmente por el anuncio de una nueva versión del MS-DOS o una mejora como ventanas. La mejor forma de mantenerse al día en cuanto a los programas disponibles es leer las revistas que atienden al mercado del MS-DOS.

JUEGOS Y ENTRETENIMIENTOS

Puesto que el MS-DOS se suele ejecutar en computadoras de negocios, Microsoft no proporcionó muchos recursos para juegos. Como sistema operativo, el MS-DOS no soporta gráficos; cualquier gráfico visualizado por un

programa tiene que ser adaptado a cada computadora. Sin embargo, esto no ha impedido a las compañías de software que hacen juegos, que conviertan dichos juegos al MS-DOS.

Como muchos de los programas de este capítulo, la mayor parte de los juegos requieren un IBM PC, o una computadora con un funcionamiento prácticamente idéntico al del IBM. La mayoría de los fabricantes han decidido escribir juegos que sólo se pueden ejecutar en un IBM PC o en computadoras de funcionamiento similar, ignorando el resto del mercado MS-DOS. En realidad, la mayor parte de los juegos del tipo de los de salón recreativo o juegos de arcada, mencionados en esta sección, requieren que el IBM PC tenga un adaptador de gráficos de color en vez de un monitor monocromo. (Algunos de los juegos que se mencionan en esta sección no son realmente juegos MS-DOS, puesto que contienen su propio sistema operativo, pero, de todos modos, se tratan aquí).

Hay una gran variedad de tipos de juegos. Los juegos tratados en esta sección se han clasificado en juegos de salón recreativo y juegos para pensar. A mucha gente no le gustan los juegos de acción de los salones recreativos debido a su violencia y a su naturaleza repetitiva. Sin embargo, es fácil evitar los juegos de disparar (o comer) a diestra y siniestra. Si no está interesado en los juegos de salón recreativo, puede que encuentre interesante algunos de los juegos para pensar.

Si usted es nuevo en los juegos de acción en computadoras, tenga en cuenta los siguientes puntos:

- No juzgue un juego por su cubierta. El arte llamativo del paquete puede parecer importante, pero recuerde que es más barato producir un paquete vistoso que pagar buenos programadores. Los programadores gráficos buenos son más caros que los artistas gráficos buenos.
- No espere más de lo que haya visto en los salones recreativos. (En realidad, probablemente usted debiera esperar muho menos). Si no le gustan los juegos de los salones recreativos, seguro que no le gustarán los juegos del mismo tipo, disponibles para el IBM PC. La mayoría de los juegos para el MS-DOS son copias conceptuales de los juegos de los salones recreativos.
- Cuente con añadir una palanca de juego (joystick) a su computadora. Muchos de los juegos que requieren velocidad y agilidad son extremadamente difíciles de jugar sin una palanca de juego. Usar las teclas de control del cursor puede ser muy difícil con estos juegos.
- Si es posible, pruebe un juego antes de comprarlo. Sin embargo, la mayor parte de las tiendas de computadoras no le permitirán probar los juegos en la tienda —especialmente en las computadoras de negocios.
- Si quiere buenos juegos de acción, quizá descubra que lo mejor que puede hacer es comprar una computadora doméstica barata (como un Apple, Atari o Commodore). Si esto es una opción práctica para usted, probablemente estará mucho más satisfecho con los resultados.

Juegos tipo salón recreativo

La mayoría de los juegos de este tipo que funcionan con el MS-DOS no son terriblemente excitantes, pero son populares. Ejemplos de juegos populares que han sido adaptados al IBM PC son BurgerTime, de Mattel Electronics (adaptación directa de la versión para salones recreativos), *Styx,* de Windmill Software (copia exacta de QIX; muy bueno), *Digger,* también de Windmill (una imitación bastante buena de DigDug), *PC-MAN,* de Orion Software (puede adivinar de que es copia) y *J-Bird,* también de Orion (una buena imitación de Q-Bert).

Otros juegos similares a los de salón recreativo son *St. Hippolyte's Wall,* de XOR (un interesante juego de muro), *Mousekattack,* de Sierra On-Line (un giro humorístico sobre el juego normal del laberinto), *Lasercycle,* de Brady (otro juego más de muros) y *Night Stalker,* de Mattel (muchos disparos y carreras). Penguin Software, Electronic Arts, Broderbund y Datasoft también tienen excelentes series de juegos para el IBM PC. *The PC Arcade,* de FriendlySoft, es una mezcla interesante de juegos de acción y juegos de estrategia; todos se pueden ejecutar tanto en monitores gráficos como en monitores de texto exclusivamente.

Algunos juegos de habilidad también son buenas herramientas para aprendizaje. Muchos de los paquetes de simulación de vuelos (tales como el de Microsoft) hacen que usted preste atención a los numerosos indicadores y luces que tiene normalmente un pequeño avión y que responda utilizando los mandos semejantes a los de un avión real. Aunque usted no obtiene puntos, como ocurre normalmente en los juegos, el objetivo consiste en despegar y aterrizar sin estrellarse.

Un juego innovador es *Night Mission Pinball,* de subLogic. Aunque hay otras simulaciones de billarines, este paquete le permite diseñar su propio campo de juego; puede añadir parachoques u objetivos, o simplemente moverlos para jugar mejor. Esto hace que sea un interesante programa de diseño en vez de sólo un juego de salón recreativo.

Juegos para pensar

Es difícil comparar los juegos para pensar con algo familiar. Algunos juegos se describen como de «ficción interactiva», mientras otros son «juegos de aprendizaje». Son similares a las historias en las que usted puede participar describiendo adónde va el personaje principal y qué es lo que hace.

Los juegos de este tipo más interesantes, son, con mucho, los de aventuras. En un juego de aventuras, usted resuelve misterios y consigue tesoros utilizando palabras para moverse por un entorno. Por ejemplo, si el programa le dice que está próximo a una cueva y usted dice que quiere entrar en ella, el programa le dirá entonces lo que encuentra allí. Puede hacer muchas cosas (conseguir tesoros, matar dragones, etc.), pero el objeto del juego es resolver un misterio o alcanzar una meta.

En el mundo de los juegos de aventuras, los juegos de Infocom están considerados como unos de los más interesantes (incluyen las series *Zork*, Planetfall y *Starcross*). Los escenarios varían desde fantasías de calabozos y dragones hasta ciencia ficción o historias normales de detectives. Los juegos de Infocom son conocidos por su humor y por sus giros intrigantes. Su interacción con el juego es sorprendentemente realista, porque puede hacer preguntas y dar listas de órdenes en inglés.

Otro juego interesante en el género de aventura y misterio es *Mystrix*, de Insoft. Este juego es como un libro de detectives interactivo donde usted dispone de opciones acerca de cómo reunir pistas. Algunos juegos de aventuras visualizan imágenes mientras usted se mueve a través del mundo de la fantasía. *Ulysses and the Golden Fleece*, de Sierra On-Line, tiene un argumento absorbente (la bien conocida leyenda) y muy buenos gráficos.

Si le gustan los juegos de palabras, hay una adaptación excelente del estándar *Jotto*, de Word Associates. Insoft tiene dos juegos de palabras, *Wordtrix* (similar a Boggle) y *Quotrix* (un juego de adivinanzas). Computer Advanced Ideas también tiene juegos similares.

Si le gustan los juegos de estrategia, puede que le guste *Pits and Stones*, de Orion Software. Es una adaptación directa del juego clásico Oh-wah-ree, y su animación es muy buena, tanto en monitores en color como en monocromos. Puede jugar contra otra persona o contra la computadora, que es un duro contrincante. *Edo*, de XOR Software, es un juego de estrategia único, con similaridades con el juego japonés Go; es difícil no volverse adicto a este juego. Bluebush tiene una excelente versión de ajedrez que la mayoría de la gente encontrará muy emocionante.

LENGUAJES DE PROGRAMACION

Este libro ha insistido en que usted no necesita aprender a programar para usar su computadora MS-DOS. Esto es debido a que la mayor parte de las aplicaciones que probablemente usted va a querer ya están escritas. Sin embargo, si se siente inclinado a ello, puede ganar mucho aprendiendo a usar diferentes lenguajes de programación. Aprender un lenguaje como BASIC puede beneficiarle de varias maneras:

- Conseguirá un mejor entendiendo de cómo su computadora realiza tareas.
- Empezará a darse cuenta de lo complicado que es programar un paquete grande, tal como un procesador de texto.
- Entenderá mejor la relación entre las diferentes componentes de su computadora (como el teclado y la CPU).

Es importante recordar que no necesita aprender nada de lenguajes para usar su computadora. Mucha gente del campo de las computadoras recomiendan el aprendizaje de un lenguaje, pero muchos otros piensan que es contraproducente para la mayoría de los usuarios. La mayor parte de la gente que aprende lenguajes nunca escribe más de uno o dos pequeños programas con ellos. Si no está interesado en aprender más acerca de los lenguajes de programación, pase por alto la próxima sección.

Aprendizaje de un lenguaje de programación

Casi todos los lenguajes de programación importantes han sido transferidos a las computadoras MS-DOS. Uno de los primeros lenguajes para microcomputadoras, BASIC, es aún el lenguaje más popular usado con el MS-DOS. En realidad, la mayor parte de los fabricantes de computadoras incluyen una copia del BASIC de Microsoft (frecuentemente denominado GWBASIC o BASICA) con el MS-DOS. Otros lenguajes disponibles son FORTRAN, COBOL, PASCAL y *lenguaje ensamblador* (también conocido como lenguaje máquina). Si no ha aprendido nunca un lenguaje de programación, probablemente debería comenzar con BASIC.

Si sabe BASIC u otro lenguaje de programación no tendrá problema en aprender GWBASIC. Algunos «maníacos de las computadoras» puede que le digan que BASIC no es realmente un lenguaje útil. Sin embargo, teniendo en cuenta cuantos juegos, paquetes de contabilidad, paquetes de comunicaciones y procesadores de texto han sido escritos en BASIC, es difícil tomar esta afirmación literalmente. Se puede argüir que algún otro lenguaje es mejor para programar, pero no que el BASIC no es útil.

La forma más fácil de aprender BASIC es leer un libro que le permita probar ejemplos mientras aprende. Compruebe que el libro que va a leer está adaptado para los usuarios del GWBASIC, puesto que hay muchos dialectos del lenguaje BASIC. La mayor parte de estos libros tienen ejemplos cortos que usted introduce mientras lee, de forma que puede comprobar sus progresos.

Si decide aprender un lenguaje diferente, debiera concentrarse en uno de los denominados lenguajes de alto nivel, como FORTRAN o COBOL. Aunque estos lenguajes no son tan útiles para programar, hay muchos buenos libros educativos para ellos (ver capítulo 7). Después de aprender un lenguaje, aprender un segundo lenguaje lleva relativamente poco tiempo.

De los lenguajes de programación, el lenguaje ensamblador es, con mucho, el más difícil de aprender. Esto es una pena, porque cuando se entienden los conceptos que hay detrás de la programación en lenguaje ensamblador, se entiende la mayor parte de lo que hace funcionar a la computadora. También es extremadamente difícil programar en lenguaje ensamblador.

Si decide programar en un lenguaje que no sea GWBASIC, tendrá una amplia gama de posibilidades. Esta sección menciona muchas de las opciones y da los nombres de sus fabricantes. Aunque hay docenas de fabricantes que no se mencionan, casi todos los lenguajes sí han sido mencionados.

La mayor parte de los lenguajes mencionados aquí son *compiladores,* esto es, lenguajes que convierten su programa a instrucciones máquina que su computadora entiende. Alguno de los lenguajes son *intérpretes,* que actúan como traductores entre su programa y la computadora. La diferencia entre los dos es que los compiladores producen ficheros que pueden ser ejecutados como las órdenes (éstos tienen extensiones de fichero EXE y COM, igual que las órdenes externas del MS-DOS), mientras los intérpretes producen ficheros que se pueden ejecutar sólo mientras se ejecuta el intérprete. Los programas compilados también se ejecutan más rápidamente que los programas interpretados. Por ejemplo, GWBASIC es un intérprete no un compilador.

BASIC

La mayoría de los paquetes BASIC populares son intérpretes en vez de compiladores. Hay otros intérpretes BASIC además del de Microsoft. Estas otras versiones de BASIC suelen tener características que GWBASIC no tiene. Si usted escribe programas grandes, *Waterloo microBasic,* de Watsoft, es más fácil de usar. Digital Research ofrece *CBASIC-86,* que es similar a su popular versión de BASIC en CP/M. También comercializa *Personal Basic,* que es más fácil de usar que GWBASIC porque verifica las órdenes mientras usted las introduce. *metaBASIC,* de Software 128, es otra buena versión de BASIC.

Microsoft, SuperSoft y Digital Research también tienen compiladores de BASIC. Estos compiladores producen programas que se ejecutan mucho más rápido que los intérpretes, pero suelen tener limitaciones que los intérpretes no tienen (como el tratamiento de errores que es más difícil de entender).

Puesto que GWBASIC es tan popular, hay muchos otros productos que puede comprar, que le ayudan a programar con GWBASIC. *Active Trace,* de Awareco, es un conjunto de utilidades que le ayude a encontrar problemas en sus programas BASIC. Es muy fácil de usar y el manual está escrito humorísticamente. Hay muchos programas similares de «referencias-cruzadas», pero éste está mejor organizado y es más útil que la mayoría.

Otro paquete excelente es el *BASIC Development System,* de Softool Systems. Este paquete permite ejecutar los programas orden a orden, de manera que es mucho más sencillo encontrar errores de programación. Incluye otras funciones que son útiles al escribir y revisar programas.

En general, usted debiera usar un intérprete BASIC por un tiempo antes de comprar un compilador (con BASIC, usted también debiera observar que los compiladores son más caros que los intérpretes). Si está usando el compilador IBM BASIC en un IBM PC y tiene un chip 8087 en su computadora (un chip añadido poco común), puede usar *87 BASIC,* de MicroWare, que hace que el compilador utilice el 8087 para un procesamiento matemático mucho más rápido.

El lenguaje C

El lenguaje C se está convirtiendo rápidamente en uno de los lenguajes de programación más populares. Aunque no es tan fácil de aprender como el BASIC, tiene muchas características que le hacen mejor, en general, para escribir programas. Por ejemplo, es relativamente fácil escribir programas coherentes en C. Desgraciadamente, algunos aspectos del lenguaje, tales como su forma de manejar cadenas de caracteres, lo hacen aún muy difícil de usar.

Todos los paquetes C mencionados aquí son compiladores. En general, los compiladores C son mucho menos caros que los compiladores BASIC. Puesto que hay un estándar bien establecido de lo que un compilador C debe hacer, convertir programas C escritos para otros sistemas operativos al sistema MS-DOS debe ser mucho más fácil que convertir programas BASIC.

Uno de los mejores compiladores C es el paquete *DeSmet C,* de C Ware Corporation. El compilador produce programas pequeños y muy rápidos y está extremadamente próximo al estándar comúnmente aceptado para compiladores C «Kernighan and Ritchie». DeSmet C es también más barato que la mayor parte de los compiladores C. El software viene con un editor muy bueno y barato (que fue también utilizado para escribir parte de este libro). C Ware también vende el mismo compilador C para CP/M-86.

Otro compilador C bien estructurado es *C86,* de Computer Innovations. Este paquete incluye el código fuente para la mayor parte de las rutinas internas, de forma que los usuarios avanzados pueden reconfigurar el paquete para adecuarlo a sus propias necesidades. La última versión presenta un *compilador optimizador* que sabe cómo hacer que los programas se ejecuten más rápido (y ocupen menos espacio en el disco).

Usualmente, es muy difícil hacer que un compilador ejecute un programa paso a paso; esto sólo suele ser posible con un intérprete. Sin embargo, c-systems ha desarrollado un compilador y depurador de C (denominado *c-window*) que permite hacerlo. Esta característica avanzada hace que sea mucho más fácil el escribir programas largos, puesto que se pueden encontrar los errores mucho más rápidamente.

Otras compañías con compiladores C para MS-DOS son Microsoft, Digital Research, Vandata, Telecon, Mark Williams Co., Lifeboat Associates y SuperSoft.

Puesto que C es tan popular, varios fabricantes venden paquetes que hacen que el trabajar con C sea aún más fácil. Por ejemplo, *Entry System for Programmers (ESP),* de Bellesoft, hace mucho más fácil editar en C porque conoce la estructura del lenguaje C. Por ejemplo, cuando usted escribe el nombre de una función, ESP sabe qué tipo de argumentos lleva dicha función y automáticamente incluye los paréntesis para la función y le dice qué es lo que se espera que introduzca.

Hay también muchos conjuntos de «herramientas C», que son subprogramas para incorporar en sus programas C. XOR vende un buen paquete, deno-

minado simplemente *C Tools;* incluye rutinas para hacer gráficos, sonido y entrada de datos en el IBM PC.

Pascal

Pascal es otro lenguaje popular que se utiliza en una gran variedad de áreas de programación. Tiene muchas características que lo hacen fácil de usar, pero los programas que produce suelen ser más largos y más lentos que los programas C. Su principal ventaja es que cuando se escriben programas en Pascal, se puede ver fácilmente la estructura de lo que se ha escrito; esto suele conducir a la realización de mejores programas. Existen muchos buenos compiladores e intérpretes de Pascal, como los de Digital Research, Microsoft y Watsoft. Modula, de Logitech, es un nuevo lenguaje similar a Pascal.

FORTRAN

FORTRAN es el lenguaje que los científicos comenzaron a usar en los años 60. Tiene pocas ventajas sobre otros lenguajes disponibles para el MS-DOS, puesto que sus mejores características fueron utilizadas por los que desarrollaron nuevos lenguajes de programación. Sin embargo, si usted ya programa en FORTRAN, quizá no quiera perder tiempo aprendiendo otro lenguaje. Hay muchos compiladores FORTRAN para MS-DOS, como los de Microsoft y SuperSoft.

COBOL

COBOL es como FORTRAN. Es un lenguaje antiguo que ha sido reemplazado por lenguajes más rápidos y más fáciles de usar. Sin embargo, los programas COBOL son aún más difíciles de escribir que los programas FORTRAN, puesto que usted frecuentemente tiene que escribir muchas más líneas de programa para producir el mismo resultado. Puesto que muchos programas están escritos en COBOL, parece que éste permanecerá durante algún tiempo al menos. mbp tiene un compilador de COBOL excelente que proporciona muchas características de las que carece el COBOL en computadoras de gran tamaño. Otras compañías que disponen de COBOL para el MS-DOS son Digital Research, Microsoft y Ryan-McFarland.

Logo

Logo es un lenguaje que actualmente es muy popular como un medio para enseñar a los niños a programar. Las ventajas de utilizar Logo con los niños

constituyen un tema que se debate con vehemencia. Logo es un lenguaje fácil de aprender que permite a los estudiantes descubrir por ellos mismos las ideas de la programación. Logo utiliza un gráfico «la tortuga», que dibuja líneas en la pantalla mientras se mueve, según las instrucciones de programación. Es mucho más fácil dibujar imágenes y modelos con Logo que con otros lenguajes.

Hay muchas versiones diferentes de Logo. *PC Logo,* de Harvard Associates, es un paquete muy completo con un manual extenso. Aunque puede que sea demasiado avanzado para niños pequeños, es más amplio que otros. *Waterloo Logo* trae un manual muy corto, lo que significa que el usuario tiene que hacer más experimentación. Sin embargo, Waterloo Logo permite utilizar sonido (PC Logo no), lo cual hace que sea más interesante que utilizar exclusivamente medios visuales.

FORTH

La gente que programa en FORTH tiene entera confianza en él; dicen que es el lenguaje del futuro y que todo el mundo debiera aprenderlo. FORTH utiliza un método (sintaxis) completamente diferente de dar órdenes y es a menudo incomprensible para la mayoría de los programadores; la mayor parte de los usuarios piensan que suele ser difícil aprender a usar FORTH. Los programadores FORTH suelen ejecutarse más rápidamente que los programas en otros lenguajes y ocupar menos espacio en RAM. Si decide aprender FORTH, hay unos cuantos buenos libros sobre el tema.

Muchas versiones de FORTH se pueden utilizar con el MS-DOS. Una de las mejores es *PC/FORTH,* de Laboratory Microsystems. El programa es fácil de instalar y la documentación es excelente. (Esto es importante para un lenguaje como FORTH, puesto que suele ser muy difícil descifrar cómo usarlo). Otras versiones de FORTH para MS-DOS son las de Quest Research, Satellite Software International, y Mountain View Press.

Ada

En 1983, el Departamento de Defensa de los Estados Unidos de América, designó oficialmente a Ada como el lenguaje que utilizará dicho departamento pàra desarrollar programas. Ada es similar a Pascal, pero tiene unas reglas mucho más rigurosas de cómo escribir programas. Aunque un usuario ocasional probablemente no estará interesado en ello, RR Software ha desarrollado un subconjunto de Ada llamado *Janus/Ada* para el MS-DOS. Este paquete está muy bien documentado, aunque la cantidad innumerable de reglas de programación en Ada echará para atrás a la mayoría de los programadores.

Pilot

Pilot es un lenguaje interesante, porque no se usa para escribir programas. En cambio, se utiliza para escribir tutoriales. Sirve para que los profesores puedan juntar materiales educativos utilizados en computadoras. *PC/Pilot* es fácil de usar y está disponible en Wasthington Computer Services.

APL

Si tiene problemas que implican el uso de matrices matemáticas, APL es el mejor lenguaje para programar las soluciones. Su sintaxis es aún más extraña que la de FORTH's, requiriendo que usted utilice en su teclado y pantalla símbolos que no son del alfabeto estándar. Sin embargo, el inconveniente de aprender un lenguaje extraño se compensa con el magnífico tratamiento de matrices que proporciona el APL. El paquete APL más popular para el IBM PC es *APL*Plus/PC,* de STSC. Es un APL completo y trae unos cuantos manuales bien escritos.

LISP

La *inteligencia artificial (AI)* se está convirtiendo en un tema de investigación muy popular, especialmente entre los fabricantes japoneses de computadoras. En realidad, algunos de estos fabricantes, como Fujitsu, están implementando los resultados de la investigación americana en AI mucho más rápidamente que las empresas americanas. Muchos programas que usan AI están poniéndose a la venta para usuarios de microcomputadoras. El lenguaje más utilizado por los investigadores en inteligencia artificial es LISP. Igual que FORTH y APL, LISP tiene un formato de programación no tradicional que echa para atrás a la mayoría de los usuarios principiantes. La teoría actual de la AI es que LISP hace un buen trabajo modelando muchos procesos del pensamiento humano.

Norell Data Systems tiene un intérprete LISP muy bueno, con un manual bastante bueno. Otros fabricantes, como Microsoft y Soft Warehouse también disponen de LISP.

OTROS PROGRAMAS

Muchos programas no encajan en categorías bien definidas. A pesar de todo, algunos de estos programas son más interesantes y útiles que muchos otros mencionados en las secciones precedentes. Esta sección explora los más irnportantes de estos programas.

Programas de seguridad de los datos

La seguridad de los datos es un problema serio en las computadoras personales. Muchas compañías tienen datos (tales como listas de clientes e informes de fabricación) que podrían ayudar a las compañías competidoras a echarles del negocio. Si usted transmite datos susceptibles por líneas telefónicas, o incluso en disquetes, hay aún más posibilidades de que sus datos puedan ser interceptados. En estos casos, se necesita un método para *cifrar* los datos (hacerlos ilegibles a los intrusos). El método de cifrado apoyado por el National Bureau of Standards se llama Data Encryption Standard (DES). Tenga en cuenta, sin embargo, que muchos expertos en seguridad cuestionan la utilidad a largo plazo del DES y piensan que incluso puede que no sea seguro ahora.

DATASAFE, de IMSI, es un programa simple que permite usar el DES sin añadir ningún hardware especial al sistema. Simplemente se ejecuta un programa sobre un fichero que se quiere cifrar. El programa es muy rápido, convirtiendo alrededor de 1K de texto por segundo. El manual incluye una buena discusión sobre seguridad. Para descifrar el mensaje, se da al programa el fichero cifrado y el código secreto que se usó para cifrarlo.

Otra técnica excelente de cifrado, llamada de *clave pública,* se está haciendo muy popular, a pesar (o a causa) de la desaprobación gubernamental. Este método permite mucha más flexibilidad en la forma de transmitir mensajes y no requiere que ningún código secreto sea transmitido. *Crypt Master,* de Digital Signature, utiliza el sistema de clave pública.

Programas estadísticos

Muchas compañías reúnen datos de sus clientes, pero a menudo hacen muy poco análisis de dichos datos, debido a la falta de software. Sin embargo, hay buenos programas estadísticos disponibles para el MS-DOS, que son bastante fáciles de usar. Uno de éstos, *ABSTAT* de Anderson-Bell, es bueno para producir informes estadísticos y tiene una excelente posibilidad de ayuda, construida dentro del software. *MathStat,* de MPR/Mathematica, tiene un manual muy bueno que explica todas las estadísticas que el programa produce, así como sus aplicaciones. Otro paquete estadístico es *MicroStat* de ECOsoft.

Programas que potencian el teclado

Muchos paquetes de aplicaciones ignoran las curvas de aprendizaje de sus usuarios. Por ejemplo, cuando usted acaba de empezar, puede que necesite pensar en cada orden mientras la introduce, pero después de un tiempo, el pulsar las mismas teclas cada vez se vuelve muy aburrido. *ProKey,* de Rose-Soft, permite redefinir las teclas del teclado, de forma que al pulsar una tecla

se obtenga el mismo efecto que pulsando muchas. (Esto se llama *reconfigurar* las teclas). ProKey ahorra una inmensa cantidad de mecanografía y hace que casi todos los programas de aplicaciones sean más fáciles de usar. Por ejemplo, si su procesador de texto requiere que usted pulse cuatro series diferentes de teclas cada vez que quiere realizar una función, ProKey puede hacer que una de las teclas de su teclado produzca las cuatro series.

ProKey es también una gran ayuda para la gente con incapacidades físicas, puesto que simplifica las pulsaciones que necesitan dar. Por ejemplo, les permite no tener que pulsar la tecla CONTROL al mismo tiempo que otras teclas. *SmartKey II,* de Heritage Software, es similar a ProKey, pero no tiene tantas características.

Resolución de fórmulas

Si trabaja con fórmulas matemáticas le gustará *TK!Solver,* de Software Arts. Este programa innovador (de los smos que desarrollaron originalmente VisiCalc) permite introducir ecuaciones y variables y, a continuación, deja que la computadora utilice el álgebra para despejar las incógnitas. Por ejemplo, puede dar una complicada ecuación de dinámica de fluidos y algunas de las variables y pedir la solución; TK!Solver le dirá lo que necesita para poder encontrar la respuesta. También convertirá las unidades que haya en las ecuaciones.

Productividad personal

Dayflo (de Dayflo Software) es un interesante cruce entre un sistema procesador de texto y un DBMS. Dayflo se utiliza para introducir memorias, cartas, listas de clientes, etc. Igual que un DBMS, Dayflo se mantiene al tanto de los campos, tales como nombre y número de teléfono, de forma que usted puede pedir que se visualice toda la información que se haya introducido, relativa a una determinada persona. A continuación, el programa le mostrará cada detalle correspondiente a esa persona aunque los datos vengan desde diferentes áreas de documentos hasta su disco. Esto significa que usted no tiene que mantenerse al tanto de lo que hace cada base de datos: Dayflo las consulta todas a la vez.

Si prepara un guión antes de comenzar a escribir un informe o una memoria, *Think Tank,* de Living Videotext, Inc., le ayudará a organizar sus pensamientos. Cuando se prepara un guión con Think Tank, se puede fijar la atención en un nivel determinado de información o excluir todos los niveles más bajos. Esto le ayuda a preparar el guión secuencialmente. También se puede usar como un procesador de texto, de bajo nivel, puesto que se pueden añadir párrafos al guión y extraer dichos párrafos más tarde.

Herramientas Software

Hay paquetes que proporcionan un cierto número de pequeños programas que usted siempre ha deseado, tales como uno que mueve columnas en un fichero texto u otro que cuenta palabras en un fichero. Un ejemplo excelente de estos paquetes es *Software Tools* de Carousel Microtools. Este paquete incluye muchas herramientas para ficheros texto, así como otras utilidades que ayudan a gestionar ficheros y directorios.

Declaración de impuestos

Cuando no use la computadora para los negocios de la compañía naturalmente puede utilizarla para sus propios asuntos. *MicroTax* (de Microcomputer Taxsystems, Inc.) es un programa muy popular para declarar los impuestos personales sobre la renta y para ayudar a planificar futuros impuestos. Está bien documentado y tiene en el interior del programa todas las leyes (americanas) aplicables en cuestión de impuestos. Incluso imprime los formularios legalmente aprobados (en USA). Aunque está orientado a gestorías que hacen declaraciones de impuestos, también es útil para personas con declaraciones complicadas.

Ayudas para la programación

La mayor parte de la gente que programa en lenguaje ensamblador han encontrado su salvación en *DEBUG,* el depurador simbólico que viene con el MS-DOS. Sin embargo, hay un paquete excelente llamado *CodeSmith-86,* de Visual Age, que proporciona muchas más funciones que DEBUG, además de incorporar las mejores características de un editor orientado a pantalla. CodeSmith-86 hace la depuración mucho más sencilla.

7

OTRAS PUBLICACIONES SOBRE EL MS-DOS

Revistas de computadoras
Libros de computadoras

Una de las mejores formas de continuar aprendiendo todo lo relativo a su computadora y al MS-DOS es leer libros y revistas. Cuando busque libros y revistas de computadoras observará que hay docenas para elegir. Este capítulo le indica cómo ahorrar tiempo y dinero construyendo una colección de referencias.

La industria editorial se ha superado a sí misma publicando libros para el mercado de las computadoras. Hay generalmente dos categorías de libros: los editados en rústica, que suelen costar por debajo de 8 $ y los libros especializados, que cuestan normalmente entre 15 $ y 25 $. Las revistas van desde *Popular Computing,* que cubre ligeramente todo el rango de computadoras personales, hasta *BYTE,* que es muy técnica y a menudo supera las 700 páginas.

Es casi tan fácil «quemarse» con las publicaciones de computadoras como con las propias computadoras. Con tantos libros para elegir (y tantos que están ya anticuados) y revistas a las que susbribirse (muchas de las cuales puede que no traten sus temas de interés), usted podría darse por vencido. Este capítulo le ayudará a concentrar su dinero y su tiempo en las publicaciones adecuadas.

REVISTAS DE COMPUTADORAS

Las revistas de computadoras tienen una ventaja inherente sobre los libros: debido a que suelen ser actuales, los cambios en el mercado se reflejan rápidamente en su contenido. La plétora de anuncios puede ayudarle a saber rápi-

BYTE 70 Main Street Peterborough, NH 03458	*Personal Computer Age* 1981 Locust Street Pasadena, CA 91107
Info World 1060 Marsh Road Menlo Park, Ca 94025	*Popular Computing* 70 Main Street Peterborough, NH 03458
PC Magazine One Park Avenue New York, NY 10016	*Programmer's Journal* 2765 Potter Street Eugene, OR 97405
PC Tech Journal The World Trade Center, Suite 211 Baltimore, MD 21202	*Softalk for the IBM PC* Box 60 North Hollywood, CA 91603
PC World 555 De Haro Street San Francisco, CA 94107	

Figura 7-1. Nombres y direcciones de revistas

damente lo que está disponible. Por otra parte, las revistas de computadoras pueden, desde luego, intimidar a causa de la cantidad de información que hay en cada número.

Hay por encima de 50 revistas sobre microcomputadoras, muchas de las cuales cubren todo el campo de las microcomputadoras. De las numerosas revistas para el IBM PC y los PC «junior», unas son para principiantes y otras para usuarios avanzados. La mayoría de las revistas para IBM PC son igualmente apropiadas para los que tienen computadoras similares al PC. Sin embargo, si su computadora MS-DOS no es muy compatible con el PC, no hay mucho en estas revistas que le pueda interesar. La figura 7-1 lista los nombres y direcciones de las revistas tratadas en esta sección.

Algunas de las revistas generales sobre microcomputadoras contienen información interesante e importante para los usuarios del MS-DOS. Por ejemplo, la información relativa a las tendencias de la industria de las computadoras es importante porque, desde luego, afectará al futuro de las computadoras MS-DOS.

Una buena estrategia para mantenerse informado es suscribirse a una o dos de las revistas relacionadas con IBM (especialmente si usted posee una computadora compatible con IBM) y a una de las revistas generales sobre microcomputadoras. Recuerde que usted no está obligado a leerlo todo en estas revistas; en realidad, es probable que encuentre que muchos de los artículos no son ni relevantes ni interesantes.

Revistas sobre el IBM PC

De las muchas revistas sobre el IBM PC, tres destacan como especialmente informativas. Son *Softalk, PC World* y *PC Magazine.* Cada una tiene un estilo diferente y un contenido diferente. Probablemente debiera comprar un ejemplar de cada, hojearlos y elegir una para suscribirse a ella.

Softalk for the IBM Personal Computer (también hay *Softalk for the Apple*) es una estupenda revista que no se toma a sí misma demasiado seriamente. Si posee un IBM PC, puede obtener gratis la suscripción del primer año, simplemente mandando al editor el número de serie de su PC. *Softalk* tiene menos páginas y, aproximadamente la mitad de la tirada de las otras dos revistas, pero tiene un tono mucho más amistoso. Cada mes hay concursos que desafían su creatividad, un estudio a fondo de alguna compañía importante de software y excelentes artículos, tanto para principiantes como para expertos. Las cartas al editor también son interesantes, más que en otras revistas de microcomputadoras.

PC World tiene más partidarios que *Softalk.* La mayoría de sus artículos están bien escritos y muchos tienen ejemplos fáciles de seguir. Aunque tiene menos páginas que *PC Magazine,* la tirada de *PC World* es aproximadamente la misma. Generalmente, esta revista llega al rango más amplio de lectores, aunque tiene menos artículos para principiantes que las otras dos. Con todo, *PC World* ha conseguido una buena reputación como revista innovadora y excitante, debido a su deseo de probar nuevas ideas. Por ejemplo, *PC World* fue la primera revista que incluyó un disquete de demostración en uno de sus números, como promoción para uno de sus anunciantes.

PC Magazine lleva más tiempo que las otras revistas sobre el PC y su tamaño lo demuestra. El volumen total de la revista (a menudo supera las 500 páginas por ejemplar), gran parte del cual son anuncios, la hace muy útil para la gente que está interesada en encontrar nueva información leyendo anuncios. Los artículos de *PC Magazine* están más orientados a los principiantes que los de *PC World;* como resultado, puede que no traten los temas con tanta profundidad como los de *Softalk* o *Pc World.* Algunas de las columnas habituales son excelentes (como las columnas dirigidas a las distintas profesiones) y la revista realiza revisiones de muchos productos cada mes. (Estas revisiones, sin embargo, a veces son menos críticas que las revisiones de otras revistas).

Otras revistas sobre el IBM PC son

- *Personal Computer Age,* revista para principiantes.
- *PC Tech Journal,* revista dirigida sólo a usuarios avanzados y publicada por los mismos que hacen *PC Magazine.*
- *Programmer's Journal,* otra revista para usuarios avanzados.

Las revistas que usted recibe cuando se hace de alguna asociación local de usuarios son, a menudo, muy buenas. Estas revistas suelen contener noticias y

artículos informativos. Las asociaciones de usuarios también proporcionan una buena oportunidad para preguntar acerca de las revistas en las que podría estar interesado en suscribirse.

Revistas sobre microcomputadoras en general

Tres de las revistas más populares sobre microcomputadoras son *BYTE*, *Popular Computing* e *InfoWorld*. Si está interesado sólo en su computadora MS-DOS y no en los otros avances de la industria de las computadoras, estas revistas puede que no le sean tan relevantes como las mencionadas en la última sección. Sin embargo, cuanto más involucrado esté en el mundo de las computadoras más necesitará seguir las tendencias generales.

BYTE es similar a *PC Magazine* en su longitud y en su abundancia de anuncios. *BYTE* es la revista «de première» para la industria informática. Muchos de los artículos son totalmente informativos, pero raramente traen noticias, debido a los cuatro meses, como mínimo, necesarios para publicar artículos. Además muchos de los artículos son sólo para usuarios avanzados, puesto que ésta es la principal audiencia de *BYTE;* sin embargo, *BYTE* tiene muchos artículos y columnas para usuarios medios y para la gente que quiere seguir la industria de las computadoras. *BYTE* a menudo trata los detalles técnicos de los principales nuevos productos mejor que otras revistas.

Popular Computing la publican los mismos que publican *BYTE*, pero está más orientada al usuario principiante. Los artículos suelen ser introductorios y muy pocos tratan productos no relacionados con los usuarios de microcomputadoras. Aunque hay menos anuncios en *Popular Computing*, la calidad de sus artículos hace que sea una buena compra. Esta revista pone especial énfasis en las aplicaciones de las microcomputadoras en los negocios.

InfoWorld es el periódico semanal de la industria. Como tal, es una de las mejores fuentes de noticias sobre los avances en la industria de las microcomputadoras. Casi la mitad de *InfoWorld* se dedica a noticias y artículos, y la otra mitad a revisiones de productos y artículos más largos acerca del estado de un campo particular (tal como los gráficos por computadora o el uso de las computadoras en las escuelas). Las revisiones son muy informativas e incluyen una tarjeta con las distintas puntuaciones, para una rápida consulta. La mayor parte de *InfoWorld* está dirigida hacia una amplia audiencia y muchos de los artículos y columnas se ocupan del impacto social de la revolución de las microcomputadoras.

Naturalmente, hay otras muchas revistas que se ocupan de las microcomputadoras. Si quiere tener una idea de su contenido, busque un quiosco de revistas con un gran número de títulos sobre computadoras y compre uno o dos nuevos títulos cada mes. Uno de los mejores sitios para buscar una buena tienda de revistas de computadoras es cerca de una universidad o de una zona industrial.

Un nuevo tipo de revista sobre el IBM PC está disponible en disquetes. Es-

tas revistas suelen tener artículos y programas que usted puede usar. Este nuevo formato tiene algunas ventajas (no necesita mecanografiar los programas tratados en la revista, y frecuentemente puede buscar un tema particular). Sin embargo, puede que encuentre estas revistas en disquetes difíciles de leer, puesto que tiene que usar la pantalla de la computadora.

Recuerde que leer revistas lleva tiempo, especialmente porque suelen tratar los mismos temas actuales. Probablemente es acertado que se conceda a sí mismo un cierto tiempo para leer revistas y, si se pasa del límite, empiece a leer más superficialmente.

Cómo aprovechar al máximo las revistas de computadoras

Las revistas de computadoras son tan distintas de otras revistas que mucha gente se siente abrumada con ellas. Si está familiarizado con las diferentes características que tienen las revistas de computadoras, usted usualmente puede obtener más información importante con menos dificultad. Por ejemplo, mucha gente gasta entre cino y diez horas con cada ejemplar mensual de *BYTE* y frecuentemente terminan exhaustos; realizar alguna planificación puede hacer más útil su tiempo de lectura.

Puede que esté acostumbrado a ignorar los anuncios en la mayoría de las revistas que lee. Si está interesado en aprovechar al máximo las revistas de computadoras, debiera romper esta costumbre, puesto que hay gran cantidad de información importante en los anuncios. Frecuentemente, de hecho, el número de páginas de anuncios iguala o supera el número de páginas de texto. Los anuncios son el mejor sitio para enterarse de nuevos productos y mejoras en productos más antiguos.

Los anuncios también son una buena forma de determinar el precio aproximado de un determinado software o hardware. Por ejemplo, muchas tiendas que hacen descuentos listan los precios de cientos de productos, haciendo muy fácil la búsqueda de precios. Como siempre, no puede creer todo lo que lea en los anuncios. Los fabricantes de hardware y software tienen una tendencia comprensible a exagerar la utilidad o la valía de sus productos; afortunadamente, la mayoría de las revistas revisan muchos de los productos que ve anunciados, de manera que usted normalmente puede llegar a saber más acerca de un producto, a partir de una fuente menos partidista.

Si una tienda de revistas local tiene revistas de computadoras, suele recibirlas una o dos semanas antes que la gente que tiene suscripciones por correo. Si es importante que obtenga su información pronto, debiera considerar la posibilidad de comprar las revistas en una tienda. Sin embargo, esto suele ser significativamente más caro que las suscripciones.

Cuando usted coge una revista, puede que quiera quitar todas las tarjetas de anuncios que van unidas a la revista; esto normalmente hace que el volver las páginas sea más sencillo y ayuda a mantener la revista abierta cuando está

leyendo un artículo. También suele ser útil tener lápiz y papel a mano, para tomar notas mientras se hojea el ejemplar.

En lugar de leer la revista página por página, algunas personas prefieren leer los artículos y columnas interesantes, separadamente de la publicidad. A algunos les gusta leer los anuncios antes de los artículos para conocer las novedades primero; otros prefieren extraer la información de los artículos antes de ser arrasados por los anuncios.

Frecuentemente se pueden mirar todos los anuncios de una revista, incluso aquellas tan grandes como *BYTE* o *PC Magazine,* en unos pocos minutos. Después de examinar los anuncios de unos cuantos números, probablemente reconocerá los anuncios que vio en el último número y los saltará rápidamente. Si un anuncio parece interesante, léalo y anote la compañía, producto y número de página para que pueda encontrarlo de nuevo posteriormente.

La mayoría de las revistas tienen un índice de anunciantes, próximo al final de la revista. Esto es especialmente útil si usted recuerda el nombre de la compañía que fabrica un producto, pero no recuerda en qué parte de la revista vio el anuncio. Algunas revistas también tienen tarjetas de servicio al lector, que puede usar para obtener más información después de leer un anuncio (el número del servicio al lector se menciona frecuentemente en el índice de anunciantes). La tarjeta de servicio al lector se proporciona como un servicio al fabricante; desgraciadamente, muchos anunciantes no contestan a las peticiones por tarjeta o tardan meses en hacerlo. Sin embargo, casi todos los fabricantes contestarán a una carta o incluso a una tarjeta postal. (Esto podría ser una estupenda aplicación para su procesador de texto).

La mayoría de las revistas también tienen una sección de «Nuevos productos» que realmente es publicidad y no editorial. Las revistas normalmente copian lo que se les envía a esta sección y no suelen cobrar a los fabricantes por este espacio. Este es a veces un buen sitio para encontrar información descriptiva sobre un producto.

Frecuentemente, usted puede decir qué artículos del índice de materias quiere leer. También es probable que quiera leer alguna de las columnas habituales que aparecen en todos los números, bien porque tratan material interesante para usted o porque los autores escriben bien. También encontrará probablemente columnas o secciones regulares que no le son útiles; sálteselas. Por ejemplo, las cartas al editor generalmente no son muy útiles, aunque a veces contienen aclaraciones importantes sobre artículos previos.

Muchas revistas publican artículos de compañías que esencialmente están anunciando sus productos. Antes de leer un artículo, mire la profesión y el lugar de trabajo del autor. Esto puede ahorrarle la lectura de un artículo entusiasta sobre un producto, para darse cuenta, al final, de que ha sido escrito por el fabricante del producto o por alguien que tiene un interés creado en el producto.

LIBROS DE COMPUTADORAS

Potencialmente, los libros son fuentes de consulta mucho mejores que las revistas, porque pueden desarrollar los temas individuales en mayor profundidad. Aunque algunas revistas tienen columnas mensuales sobre temas específicos, podría hacer falta un año para que la columna presentara tanto material como un libro podría tratar en un capítulo. Si no le gustan los anuncios, le será más fácil leer libros sin distraerse.

La industria de las microcomputadoras está cambiando rápidamente, sin embargo, e igual que los productos que describen, la mayoría de los libros de microcomputadores tienen sólo un período limitado de utilidad antes de quedarse anticuados. Considerando que el MS-DOS ni siquiera existía cinco años antes de publicar este libro, es perfectamente concebible que dentro de otros cinco años alguien podría pensar que este libro es raro, porque habla de las «características avanzadas» de la versión 2 del MS-DOS. Sin embargo, la mayoría de los libros están pensados para ser leídos y utilizados como referencia durante muchos años.

Los libros de computadoras son más caros que otros libros en rústica, igual que las revistas de computadoras son más caras que otras muchas publicaciones mensuales. Cuando esté imaginando cuánto se gastará, una suposición aproximada es que un libro le costará alrededor de las tres cuartas partes del coste de una subscripción a una revista. Si usted es un lector cuidadoso que le gusta leer bastante de un mismo tema, probablemente debiera comprar más libros que revistas; si prefiere una información superficial sobre un gran número de temas, probablemente le irá mejor suscribirse a más revistas.

Como con las revistas de computadoras, es fácil que se sienta abrumado cuando intente seleccionar libros que le ayudarán a usar la computadora. Hay muchos a la venta, pero debiera hacer la selección cuidadosamente. Las siguientes dos secciones describen unos cuantos de los mejores libros escritos sobre el MS-DOS (o el IBM PC) y sobre computadoras y software en general.

Libros sobre el MS-DOS y el IBM PC

Your IBM PC Made Easy de Jonathan Sachs (Osborne/McGraw-Hill, 1983) proporciona una vista general muy completa de mucho de lo que necesita saber sobre su IBM PC. Sach trata tanto el hardware como el software y da abundantes ejemplos de cómo usar las órdenes del MS-DOS. Hay una excelente sección sobre el hardware interno del PC, cómo cuidarlo y qué hacer en caso de problemas.

Peter Norton es un autor muy conocido que escribe muchos artículos para las revistas relacionadas con el PC. Dos de sus libros, *Inside the IBM PC: Access to Advanced Features and Programming* y *MS-DOS and PC-DOS: User's Guide* (Robert J. Brady Co., 1983) son buenas guías para usar el IBM PC. El primer libro es con mucho la mejor referencia técnica para el PC;

tiene excelentes diagramas y tablas, y trata algunos temas interesantes que faltan en otros libros avanzados. El segundo libro es demasiado técnico para algunos lectores, aunque tiene gran cantidad de información y bastantes dibujos cómicos. También tiene un maravilloso glosario.

Otro buen libro general sobre el IBM PC es *IBM Personal Computer Handbook,* editado por Dzintar Dravnieks y otros (And/Or Press, 1983). Esta es una colección de artículos de muchos autores, además de una buena lista de recursos para el PC.

Si quiere un buen conjunto de referencias organizadas por orden alfabético, vea la *Reference Encyclopedia For the IBM Personal Computer,* de Karen y Gary Phillips (Ashton-Tate, 1983). Esta enciclopedia de dos volúmenes es una excelente obra de consulta y en ella puede encontrar casi de todo, incluyendo información acerca de los centenares de paquetes software y periféricos.

Para principiantes, *Your IBM Personal Computer: Use, Applications, and BASIC,* de David Cortesi (CBS College Publishing, 1982) tiene algunas buenas discusiones sobre lo que son las computadoras y lo que pueden hacer por usted. Sin embargo, el libro no está actualizado.

Otros libros útiles

Hay otros muchos libros que no se relacionan directamente con el MS-DOS o con el IBM PC, pero que, a pesar de todo, son muy útiles. Algunos proporcionan información general sobre computadoras y sobre la industria informática; otros tratan de importantes programas de aplicación. Estos libros puede que le resulten de mayor utilidad que los libros mencionados previamente, porque proporcionan una visión en profundidad de los conocimientos de un campo.

Para la gente interesada en el impacto social de las computadoras sobre sus vidas, además de sobre las vidas de los demás, el libro *Computer Power and Human Reason: From Judgement to Calculation,* de Joseph Weizenbaum (W. H. Freeman, 1976) es una excelente discusión sobre un cierto número de temas sociales. Entre los temas se incluyen las dificultades personales con las computadoras y lo que pueden hacer realmente las computadoras. Weizenbaum es ahora famoso por sus conocimientos, y el libro se entiende fácilmente.

Si quiere saber más de procesamiento de textos, *The Word Processing Book, Questions and Answers on Word Processing* y *Word Processing on the IBM Personal Computer,* de Peter McWilliams (Prelude Press) son buenas fuentes de información. Aunque el estilo de McWilliams es a menudo coloquial, da buena información práctica y sus analogías suelen aclarar los conceptos difíciles.

Data Base Management Systems: A Guide to Microcomputer Software, de David Kruglinski (Osborne/McGraw-Hill, 1983) tiene una excelente descrip-

ción de varios DBMS y de cómo trabajan. Aunque el software descrito en el libro es para CP/M, gran parte de este material es, sin embargo, útil para el usuario del MS-DOS.

Probablemente, el libro más práctico para cuando se compra o se posee una pequeña computadora es *Computer Wimp* de John Bear (Ten Speed Press, 1983). Está lleno de maravillosas anécdotas, consejos, citas célebres y sentido común; su descarada irreverencia con los fabricantes de hardware y software es refrescante. Es un excelente libro para dar a alguien que está buscando una computadora.



LISTAS DE REFERENCIA DE ORDENES

La tabla A-1 lista las órdenes MS-DOS por orden alfabético. También proporciona el grupo funcional dentro del cual se trata cada orden en el capítulo 3. La tabla A-2 lista todas las sintaxis y argumentos de las órdenes.

Tabla A-1. Listado alfabético de las órdenes MS-DOS

Orden	Interna	Externa	Descripción	Grupo funcional
ASSIGN		*	Cambia las letras asignadas a los discos	Indicadores del sistema
BACKUP		*	Hace copias de seguridad del disco duro	Mantenimiento de discos
BASICA		*		
BREAK	*		Hace que el MS-DOS compruebe más a menudo si hay interrupciones de usuario	Indicadores del sistema
CHDIR	*		Cambia el subdirectorio por defecto	Mantenimiento de caminos
CHKDSK		*	Examina y arregla discos	Mantenimiento de discos
CLS	*		Borra la pantalla	Otras órdenes
COMMAND		*	Ejecuta otro programa	Ficheros batch
COMP		*	Compara ficheros	Mantenimiento de ficheros
COPY	*		Hace copias de ficheros	Mantenimiento de ficheros
CTTY	*		Cambia la consola a una puerta serie	Indicadores del sistema

Tabla A-1. Listado alfabético de las órdenes MS-DOS *(continuación)*

Orden	Interna	Externa	Descripción	Grupo funcional
DATE	*		Pone la fecha	Indicadores del sistema
DEBUG		*	Examina y cambia ficheros binarios	Herramientas de programación
DIR	*		Visualiza la lista de ficheros de un disco	Mantenimiento de discos
DISKCOMP		*	Compara dos disquetes	Mantenimiento de discos
DISKCOPY		*	Copia un disquete entero	Mantenimiento de discos
ECHO	*		Imprime un mensaje en la pantalla	Ficheros batch
EDLIN		*	Edita ficheros texto	Mantenimiento de ficheros
ERASE	*		Suprime ficheros de un disco	Mantenimiento de ficheros
EXE2BIN		*	Convierte ficheros EXE en ficheros COM	Herramientas de programación
FDISK		*	Realiza mantenimiento en el IBM PC/XT	Mantenimiento de discos
FIND		*	Busca texto en un fichero	Otras órdenes
FOR	*		Repite una orden para varias opciones	Ficheros batch
FORMAT		*	Prepara un disco para el MS-DOS	Mantenimiento de discos
GOTO	*		Salta a una parte distinta de un fichero batch	Ficheros batch
GRAPHICS		*	Permite imprimir pantallas gráficas	Otras órdenes
IF	*		Ejecuta órdenes según una decisión	Ficheros batch
LINK		*	Combina ficheros objeto	Herramientas de programación
MKDIR	*		Añade subdirectorios a un disco	Mantenimiento de caminos
MODE		*	Cambia los parámetros de comunicación	Indicadores del sistema
MORE		*	Visualiza un fichero en la pantalla con pausas	Salida de ficheros

Tabla A-1. Listado alfabético de las órdenes MS-DOS *(continuación)*

Orden	Interna	Externa	Descripción	Grupo funcional
PATH	*		Cambia la lista de búsqueda de órdenes	Mantenimiento de caminos
PAUSE	*		Espera a que se pulse una tecla	Ficheros batch
PRINT		*	Imprime un fichero por impresora	Salida de ficheros
PROMPT	*		Cambia la petición de orden MS-DOS	Indicadores del sistema
RECOVER		*	Repara ficheros con sectores dañados	Mantenimiento de discos
REM	*		Pone un comentario en un fichero batch	Ficheros batch
RENAME	*		Da nuevos nombres a ficheros	Mantenimiento de ficheros
RESTORE		*	Hace copias de ficheros de seguridad	Mantenimiento de ficheros
RMDIR	*		Suprime subdirectorios de un disco	Mantenimiento de caminos
SET	*		Cambia cadenas de entorno	Indicadores del sistema
SHIFT	*		Mueve los argumentos de la línea de orden	Ficheros batch
SORT		*	Ordena un fichero texto	Otras órdenes
SYS		*	Pone los ficheros del sistema y las pistas de carga inicial en un disquete	Mantenimiento de discos
TIME	*		Pone la hora	Indicadores del sistema
TREE		*	Escribe la lista de subdirectorios	Mantenimiento de caminos
TYPE	*		Visualiza un fichero en la pantalla	Salida de ficheros
VER	*		Visualiza el número de la versión MS-DOS	Indicadores del sistema
VERIFY	*		Dice al MS-DOS que haga doble comprobación cuando escriba en disco	Indicadores del sistema
VOL	*		Visualiza el rótulo de un disco	Indicadores del sistema

Tabla A-2. Sintaxis y argumentos de las órdenes MS-DOS

Orden	**Sintaxis y argumentos**
ASSIGN	ASSIGN [*d1=d2...*]
BACKUP	BACKUP *especifichero d:*
BREAK	BREAK [{ON|OFF}]
CHDIR	CHDIR [[*d:*]*camino*] CHDIR [*d:*]
CHKDSK	CHKDSK [*d:*] [/F] [/V] CHKDSK *especifichero*
CLS	CLS
COMMAND	COMMAND /C *orden*
COMP	COMP [*especifichero1 especifichero2*]
COPY	COPY [{/A|/B}]*especifichero1* [{/A|/B}] {*d:*|*especifichero2*} [{/ A|/B}] [/V] COPY [{/A|/B}]*especifichero1a* [{/A|/B}] [+ *especifichero1b*] [{/A|/B}]...[/V] {*d:*|*especifichero2*} [{/A|/B}] [/V] COPY {*especifichero1*|*dispositivo1*} {*especifichero2*| *dispositivo2*}
CTTY	CTTY *dispositivo*
DATE	DATE [*mm-dd-aa*]
DEBUG	DEBUG [*especifichero*] [*args*]
DIR	DIR [{*d:*|*especifichero*}] [/P] [/W]
DISKCOMP	DISKCOMP [*d1:*] [*d2:*] [/1] [/8]
DISKCOPY	DISCOPY [*d1:*] [*d2:*] [/1]
ECHO	ECHO [{ON|OFF|*mensaje*}]
EDLIN	EDLIN *especifichero* [/B]
ERASE	ERASE *especifichero*
EXE2BIN	EXE2BIN *especifichero* [*especifichero*]
FDISK	FDISK
FIND	FIND [/V] [/C] [/N] *"cadena"* [*especifichero...*]
FOR	FOR %%*nombrevar* IN (*lista*) DO *orden* *FOR* %%*nombrevar* IN (*especifichero*) DO *orden*
FORMAT	FORMAT [*d:*] [/S[/1] [/8] [/V] [/B]
GOTO	GOTO *rótulo*
GRAPHICS	GRAPHICS
IF	IF [NOT] *condición orden*

Tabla A-2. Sintaxis y argumentos de las órdenes MS-DOS *(continuación)*

Orden	Sintaxis y argumentos
LINK	LINK LINK @ *especifichero*
MKDIR	MKDIR [*d:*]*camino*
MODE	MODE LPT*n*: [*ancholínea*] [,[*altolínea*] [,P]] MODE *anchopantalla* [, {R | L} [,T]] MODE COM*n*: *baudios* [,*paridad* [,*bitsdatos* [,*bitsparada* [,P]]]] MODE LPT*n*:=COM*n*
MORE	*MORE* < *especifichero*
PATH	PATH [*d:*]*camino*[*d:*]*camino*...]
PAUSE	PAUSE
PRINT	PRINT [*especifichero* [/T] [/C] [/P]...]
PROMPT	PROMPT [*cadena-petición-orden*]
RECOVER	RECOVER *especifichero* RECOVER *d:*
REM	REM *comentario*
RENAME	RENAME *especifichero1 especifichero2*[.*ext2*]
RESTORE	RESTORE *d: especifichero*
RMDIR	REMDIR [*d:*]*camino*
SET	SET [*nombrevar*=[*cadena*]]
SHIFT	SHIFT
SORT	SORT [/R] [/+*n*] < *especifichero*
SYS	SYS *d:*
TIME	TIME [*hh:mm:ss.xx*]
TREE	TREE [*d:*] [/F]
TYPE	TYPE *especifichero*
VER	VER
VERIFY	VERIFY [{ON | OFF}]
VOL	VOL [*d:*]

B

DIFERENCIAS ENTRE LAS VERSIONES 1 Y 2 DEL MS-DOS

Cada nueva versión del MS-DOS incluye más órdenes y características que las inversiones previas; ésta es la principal diferencia entre sucesivas actualizaciones. Cada nueva versión del MS-DOS es compatible con versiones más antiguas. Así, casi todos los programas que funcionan con la versión 1 del MS-DOS funcionan con la versión 2; sin embargo, al revés no suele ser verdad. (Las pocas veces que los programas escritos para la versión 1 no funcionan, suele ser debido a errores realizados por el fabricante del software).

Otra diferencia entre las versiones es la cantidad de memoria que requiere cada una. La versión 2 usa más RAM porque es más grande que la versión 1. Si ve que los programas que funcionan con la versión 1 se salen de la memoria con la versión 2, probablemente tendrá que comprar más memoria para su computadora. Esto puede ser un problema en las computadoras que no permiten añadir bancos de memoria. Por esta razón, compruebe las necesidades de memoria de sus programas para cada versión del MS-DOS.

Frecuentemente, se hacen actualizaciones menores dentro de una versión del MS-DOS. Estas actualizaciones se reflejan en la parte del número de versión, a la derecha del punto decimal. Por ejemplo, la versión 1.1 es una actualización de la versión 1.0 y la versión 2.1 es una actualización de la versión 2.0. Las actualizaciones se usan para corregir errores o deficiencias en el sistema operativo; no se añaden nuevas órdenes ni características internas en estas actualizaciones.

NUEVAS CARACTERISTICAS DE LA VERSION 2

Las siguientes son las nuevas características de la versión 2 del MS-DOS:

- Directorios estructurados en árbol (descrito en el capítulo 4).
- Entrada y salida redirigidas y uso de tubos (tratado en el capítulo 4).

- Rótulos de volumen en discos (ver la orden FORMAT en el capítulo 3).
- Capacidad de usar discos duros (la versión 1 no podía utilizarlos).
- Otras características técnicas internas, incluyendo controladores de dispositivos instalables (que permiten a los fabricantes de hardware añadir periféricos a la computadora), programas de aplicación residentes (que permanecen en RAM) y un fichero de configuración que da al MS-DOS información adicional cuando se hace la carga inicial de la computadora.
- Capacidad en algunas computadoras (como el IBM PC) de incrementar el número de sectores por pista en un disquete de 8 a 9. Esto significa que los disquetes de la versión 2, escritos con 9 sectores por pista no pueden ser leídos en computadoras que funcionan con la versión 1.

NUEVAS ORDENES Y MEJORAS EN LA VERSION 2

La tabla B-1 lista las órdenes de la versión 1 que fueron revisadas en la versión 2. Esta tabla no incluye las mejoras que simplemente reflejan las características de los nuevos programas de órdenes, listados en la tabla B-2 (por ejemplo, la capacidad de la orden TYPE para utilizar nombres de caminos en la especificación de ficheros).

La tabla B-2 lista las órdenes añadidas a la versión 2 del MS-DOS y da una breve descripción de sus funciones. Todas estas órdenes se describen ampliamente en el capítulo 3.

Tabla B-1. Ordenes revisadas en la versión 2

Orden	Características añadidas
CHKDSK	Más métodos para recuperar los datos perdidos. La opción /F se añadió para que la orden no intentase automáticamente reparar un disco dañado
COMP	Capacidad de comparar múltiples ficheros
DIR	Visualiza el nombre del volumen, los subdirectorios y la cantidad de espacio que queda en el disco
FORMAT	Posibilidad de poner un rótulo de volumen en un disco con la opción /V

Tabla B-2. Ordenes añadidas a la versión 2

Orden	Descripción
ASSIGN	Cambia las letras asignadas a los discos
BACKUP	Hace copias de seguridad del disco duro
BREAK	Hace que el MS-DOS compruebe más a menudo si hay interrumpciones de usuario
CHDIR	Cambia el subdirectorio por defecto
CLS	Borra la pantalla
CTTY	Cambia la consola a una puerta serie
ECHO	Imprime un mensaje en la pantalla
FOR	Repite una orden para varias opciones
GOTO	Salta a una parte diferente de un fichero batch
IF	Ejecuta órdenes según una decisión
MKDIR	Añade subdirectorios a un disco
PATH	Cambia la lista de búsqueda de órdenes
PRINT	Imprime un fichero por impresora
PROMPT	Cambia la petición de orden del MS-DOS
RECOVER	Repara ficheros con sectores dañados
RESTORE	Hace copias de ficheros de seguridad
RMDIR	Suprime subdirectorios de un disco
SET	Cambia cadenas de entorno
SHIFT	Mueve los argumentos de la línea de orden
TREE	Escribe una lista de subdirectorios
VER	Visualiza el número de la versión MS-DOS
VEFIFY	Dice al MS-DOS que haga doble comprobación cuando escribe en disco
VOL	Visualiza el rótulo de un disco

C

COMPATIBILIDAD CON EL IBM PC

Este apéndice trata de las numerosas cuestiones acerca de la compatibilidad con la computadora IBM PC. La mayoría de los fabricantes que hacen computadoras MS-DOS las anuncian como «compatibles con IBM». Esta frase se usa demasiado y, a menudo, engañosamente. Puesto que hay muchas definiciones de la compatibilidad con IBM, los fabricantes pueden estirar el significado del término.

La compatibilidad es importante porque la mayor parte de los programas de aplicaciones diseñados para el MS-DOS, realmente han sido diseñados para el IBM PC exclusivamente. Muchas computadoras que no son IBM no pueden ejecutar dichos programas correctamente, porque no tienen todas las características IBM que el programa espera (como las teclas o el monitor de texto de IBM). Algunos programas usan más características IBM que otros; algunos requieren que las unidades de disco o la pantalla gráfica sean idénticas a las del IBM PC.

Desgraciadamente no hay normas claramente definidas para juzgar la compatibilidad. IBM, generalmente ignora a todas las demás compañías de computadoras y no respalda una norma de compatibilidad. Así, cualquier fabricante puede decir que es «compatible» sin que esté claro qué es lo que esto significa. Una computadora en la que funcionan casi todos los programas que se ejecutan en el IBM PC se suele denominar computadora similar al IBM PC.

DETERMINACION DE LA COMPATIBILIDAD

¿Cómo puede juzgar si una computadora es compatible con el IBM PC? Puede usar la siguiente lista para determinar la compatibilidad de su computadora. Ninguna de estas características aislada hace que una computadora sea compatible; en realidad algunos programas no funcionarán en máquinas distintas de IBM, incluso si reúnen todas las características mencionadas aquí.

Una computadora compatible con IBM tiene que

- Poder leer y escribir disquetes de 5 1/4 pulgadas como el PC.
- Ejecutar todas las versiones del MS-DOS con todos los cambios de IBM.
- Tener la RAM interna en las mismas direcciones que el PC.
- Usar las mismas direcciones internas que el sistema de entrada/salida en el PC.
- Permitir añadir tarjetas de hardware en ranuras de expansión idénticas a las ranuras del PC.
- Usar el mismo tipo de monitores y adaptadores gráficos que el PC.
- Tener las mismas teclas que el teclado del PC.
- Usar la misma medida de tiempo interna que el PC.
- Comunicar con los dispositivos hardware de la misma manera que el PC.

Aunque muy pocas computadoras cumplen todas estas restricciones, muchas computadoras cumplen la mayor parte.

Es imposible cuantificar la importancia de cada una de estas características. Por ejemplo, una computadora que tiene varias de estas características puede que sólo ejecute la mitad de los programas PC-DOS que ejecuta otra computadora, que tiene un conjunto distinto de características.

Otro método para medir la compatibilidad implica el usar términos estándar para clasificar cuánto software del PC puede ejecutar una computadora. Future Computing, una compañía consultora especializada en el mercado de microcomputadoras, ha desarrollado una escala de cuatro niveles para juzgar la compatibilidad. Este método para juzgar la compatibilidad se basa en la capacidad de la computadora para ejecutar software del IBM PC. Las categorías son: *compatible operacionalmente, compatible funcionalmente, compatible a nivel de datos* y *compatible a nivel de MS-DOS.*

Las dos categorías que la mayoría de los fabricantes intentan conseguir son las de compatible operacionalmente y compatible funcionalmente. Las máquinas compatibles operacionalmente ejecutan todo el software más popular; las máquinas compatibles funcionalmente ejecutan sus propias versiones del software más popular. Los otros dos niveles indican que hay muchos programas populares que no funcionan en las computadoras, debido a formatos incompatibles en los discos o diferentes configuraciones hardware o software.

Future Computing ofrece seminarios e informes publicados sobre la compatibilidad con IBM y una amplia variedad de otros temas. Puede ponerse en contacto con ellos en 900 Canyon Creek Center, Richardson, TX, 75080.

D

COMPUTADORAS QUE EJECUTAN EL MS-DOS

Este apéndice describe muchas de las computadoras que ejecutan el MS-DOS. La mayoría traen el MS-DOS como característica estándar; en otras es una opción.

TRES IMPORTANTES CUALIDADES DEL HARDWARE

Si tiene una computadora, realmente no necesita saber mucho del hardware para poder usarla, independientemente de lo que le digan los «expertos en computadoras». Sin embargo, hay tres cualidades que debiera conocer, que afectan al comportamiento del hardware: capacidad, velocidad y versatilidad.

- *Capacidad.* La capacidad se refiere a la cantidad de información que la computadora puede guardar, tanto en memoria interna como en disco. Hay dos formas de medir la capacidad: la cantidad de RAM y la cantidad y el tipo de almacenamiento en disco.
- *Velocidad.* ¿Con qué rapidez ejecuta la computadora los programas? La velocidad de su sistema depende de tres factores principales: la velocidad de procesamiento de la CPU, el tiempo de acceso a los datos en los discos y la cantidad de soporte para entrada/salida proporcionado a la CPU.

La velocidad de procesamiento de la CPU se mide en megahertzios o MHz. Para la gente que realiza cálculos numéricos con programas de aplicaciones contables o científicas, la velocidad de la CPU es a menudo muy importante. Cada modelo de CPU puede funcionar a diferentes velocidades; por ejemplo, una marca de computadora con un 8088 puede tener una CPU más rápida que otra.

El tiempo de acceso a los datos, o la velocidad de los discos, suele ser importante para la gente que usa programas de procesamiento de texto y programas contables, puesto que estos programas gastan una gran cantidad de tiempo leyendo y escribiendo información en el disco.

La cantidad de soporte de entrada/salida dado a la CPU es un tercer factor que afecta a la velocidad. Algunas computadoras usan chips especializados de soporte, tratados en el capítulo 1, que realizan tareas tales como la visualización de gráficos en pantalla. Estos chips hacen que las computadoras funcionen más de prisa, reduciendo el número de pasos que realiza la CPU principal.

La utilidad de la velocidad en alguna de estas áreas puede o no ser importante, dependiendo del tipo de trabajo que se haga. Sin embargo, si la velocidad es importante para usted, tiene que considerar varios factores al evaluarla. Estas especificaciones técnicas suele tenerlas el distribuidor de la computadora.

Puede que vea anuncios en los que los fabricantes afirman que sus computadoras son más rápidas que las de sus competidores. Estas afirmaciones deberían ser cuidadosamente examinadas, porque es fácil manipular los números relacionados con la velocidad (como, por ejemplo, mencionando que la CPU es un 10% más rápida y no mencionando que los discos son un 50% más lentos). Además, sus programas de aplicación pueden o no ser capaces de utilizar la incrementada eficiencia de la computadora. Por esta razón, una computadora rápida no es siempre la solución.

- *Versatilidad.* Una tercera cualidad que afecta al comportamiento del hardware es la versatilidad. ¿Es fácil añadir nuevo hardware a la computadora? Cuanto más tiempo hace que se tiene una computadora, más probable es que se quiera que realice tareas diferentes. Muchos fabricantes permiten incrementar la versatilidad de la computadora añadiendo nuevo hardware (esto se suele denominar *mejora*). Por ejemplo, usted podría querer añadir un disco duro, más RAM o una pantalla diferente. Si no puede añadir ningún hardware a su computadora, entonces está limitado al hardware que traía la computadora.

Hay, naturalmente, otras muchas formas de medir la calidad del hardware de la computadora, pero capacidad, velocidad y versatilidad son tres de las consideraciones más importantes. La próxima sección describe cómo se relacionan estos conceptos con el MS-DOS.



Figura D-1. Una computadora MS-DOS estándar (Eagle PC)

LINEAS DE GUIA PARA COMPRAR UNA COMPUTADORA

La mayoría de las computadoras MS-DOS parecen muy semejantes: una caja de aproximadamente 25×15×15 pulgadas con un monitor y un teclado (ver figura D-1). En realidad, muchos fabricantes se han esmerado en crear productos que se parezcan al IBM PC tanto como se pueda. Otros, intentando separarse de la mayoría de las computadoras MS-DOS, han diseñado sus computadoras para que sean portátiles. Estas máquinas suelen tener aproxi-

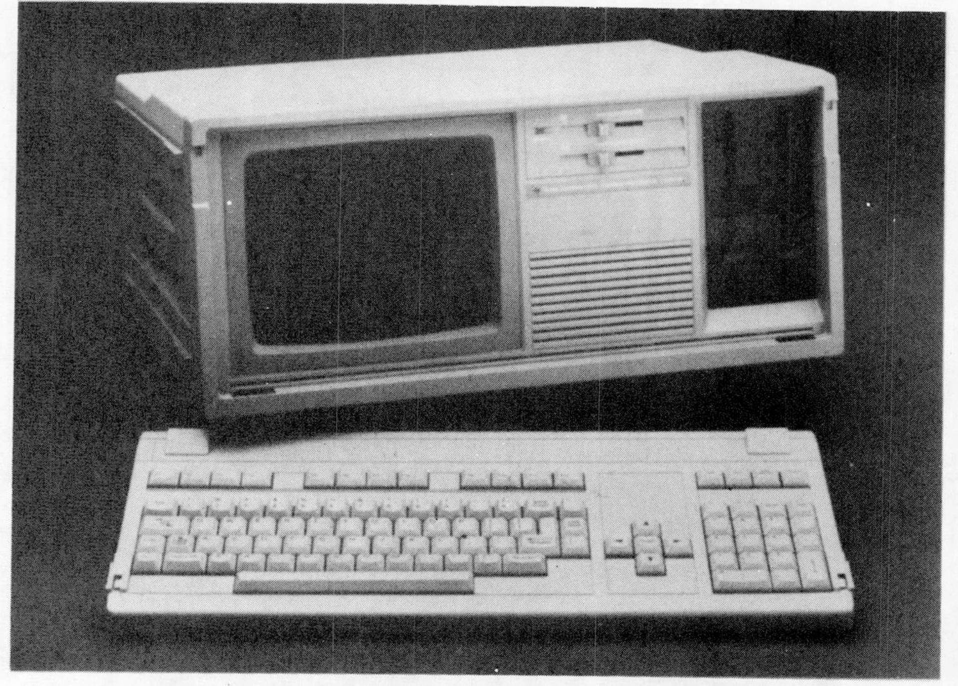


Figura D-2. Una computadora MS-DOS portátil (Texas Instruments TI Professional)



Figura D-3. Una pequeña computadora MS-DOS (Gavilan Mobile)

madamente el tamaño y la forma de una máquina de coser, y se parecen a la computadora mostrada en la figura D-2. Otras portátiles, tales como la que aparece en la figura D-3, son mucho más pequeñas, a veces suficientemente pequeñas para que cojan en una cartera.

Las tres cosas más importantes a considerar antes de comprar una computadora son la asistencia ofrecida por el distribuidor, el coste total y la longevidad de la computadora.

La asistencia ofrecida por el distribuidor debiera ser una de las primeras cosas a tener en cuenta, no sólo porque la computadora puede necesitar reparación, sino también porque puede necesitar alguna guía sobre qué hardware o software adicional comprar para su sistema.

La cantidad de ayuda que consiga al instalar su computadora es muy importante, especialmente si nunca ha tenido una computadora anteriormente. Independientemente de lo que puedan decirle los distribuidores acerca de lo fácil que son de montar, hay muchas pequeñas cosas que le frustrarán al intentar instalar su computadora correctamente. Sorprendentemente, incluso el software necesita frecuentemente ser «montado» (o, en la jerga de computadoras, *instalado*) en su sistema. Debería asegurarse de que el distribuidor también le ayudará en esto.

El coste global del sistema completo es una segunda consideración. Debería comparar los costes totales con los de otros distribuidores antes de comprar. El coste global es el coste total del hardware principal, el hardware extra añadido y todo el software necesario para satisfacer sus necesidades. Lo siguiente son algunas importantes preguntas que debe hacer:

- ¿Está incluido el monitor, el teclado o la impresora en el coste total o necesito comprarlos como accesorios?
- ¿Trae mi computadora *software incluido* (es decir, ¿qué software está incluido en el precio del sistema?) o sólo trae el MS-DOS? (O incluso, ¿está incluido el MS-DOS?)
- ¿Qué software funciona o no con el sistema? ¿Ejecutará programas PC-DOS?
- ¿Hay un coste adicional para un contrato de asistencia técnica? ¿Qué es lo que cubre la garantía?

Es difícil estimar la longevidad de una computadora (y de su fabricante), pero este factor también es importante al determinar si estará satisfecho con su computadora. Puesto que unas cuantas grandes compañías de microcomputadoras se han ido a la bancarrota en los últimos años, los distribuidores de computadoras son especialmente conscientes de las fluctuaciones de la industria. Pregunte a su distribuidor por la reputación del fabricante de la computadora. Si tiene un amigo que trabaja con microcomputadoras, puede que quiera preguntarle su opinión sobre el fabricante y el modelo que va a comprar.

La única forma de resolver estas dudas es visitar unas cuantas tiendas de computadoras y ver lo que tienen. También puede leer unas cuantas revistas populares de computadoras (como las mencionadas en el capítulo 7) para tener una idea de hacia dónde se dirige el mercado, así como para evaluar algunas de las posibilidades de las computadoras a la venta.

La siguiente sección presenta descripciones de computadoras, proporcionadas por sus fabricantes. Naturalmente, los detalles pueden cambiar al cambiar las configuraciones de las computadoras. Si está pensando en comprar una computadora, debe pedir a su distribuidor las especificaciones actuales. La información presentada aquí intenta sólo dar una visión general de las computadoras y no una comparación entre ellas.

COMPUTADORAS MS-DOS

La tabla D-1 lista algunas de las principales computadoras que ejecutan el MS-DOS; hay literalmente cientos de computadoras no incluidas aquí. La descripción de cada computadora incluye lo siguiente:

Tabla D-1. Comparación de computadoras MS-DOS

Nombre y modelo	CPU	RAM	Unidades de disco	Monitor
IBM Personal Computer (PC) y Extended Version (PC/XT)	8088	64K (PC o 128K (PC/XT)	1 (PC); 1+10M disco duro (PC/XT)	Ninguno estándar
ACT Apricot	8086	256K	Dos de 3,5 pulgadas	M 800×450
Bytec Hyperion	8088	256K	1	M 640×250

- Nombre de la computadora y modelo(s).
- CPU.
- Cantidad de RAM.
- Unidades de disco. (Son de 320K/360K y 5 1/4 pulgadas a no ser que se indique otra cosa).
- Monitor. (En la tabla, M indica monocromo y C indica color. Se menciona la resolución de los gráficos si está disponible).
- Puertas serie o paralelo (indicado con S o P en la tabla).
- Software incluido con la computadora.
- Características especiales de la computadora.
- Precio recomendado el 1 de enero de 1984.

Es importante comprobar con un distribuidor de computadoras las características de una computadora antes de comprarla. El apéndice E lista los nombres y direcciones de los fabricantes de hardware por si quiere escribir para obtener más información.

Puertas	Software incluido	Otras características	Precio
Ninguna estándar (PC); 1S (PC/XT)	BASIC	4 ranuras	$3195
1S, 1P	Sistema operativo CP/M-86 concurrente, intérprete BASIC, SuperCalc, SuperPlanner, comunicaciones, gráficos, editor de texto	Portátil, conforme a las normas ergonómicas europeas	$3190
1S, 1P	BASIC, DBMS, calculadora	Portátil, reloj	$3195

Tabla D-1. Comparación de computadoras MS-DOS *(continuación)*

Nombre y modelo	CPU	RAM	Unidades de disco	Monitor
Columbia MPC Series (VP, 1600-1, 1600-4)	8088	128K	2 (VP, 1600-1); 1+10M disco duros (1600-4)	M 640×200
COMPAQ y COMPAQ Plus	8088	128K	2 (COMPAQ); 1+10M disco duro (COMPAQ Plus)	M 640×200
Corona PC	8088	128K	1	M 640×325
DEC Rainbow 100	8088	64K	2	M
Eagle Plus, Eagle Plus XL, Eagle Spirit	8088	128K	2 (Plus y Spirit; 1+10M disco duro (Plus XL)	M 640×200
Fujitsu Micro 16	8086	128K	2	C 640×200
Gavilan Mobile	8088	64K	Una de 3,5 pulgadas (360K)	M 480×128

Puertas	Software incluido	Otras características	Precio
1S, 1P (VP); 2S, 1P (1600-1, 1600-4)	BASIC, Perfect Writer, Perfect Speller, Perfect Calc, Perfect Filer, Fast Graphs, Home Accountant Plus, Space Commanders, comunicaciones, CP/M-86, Macro Assembler, diagnósticos, disco RAM, tutorial	Portátil (VP), 8 ranuras	$2995 (VP); $3620 (1600-1); $5220 (1600-4)
1P	BASIC	Portátil, 2 ranuras	$2995 (COMPAQ); $4995 (COMPAQ Plus)
1S, 1P	BASIC, procesador de texto Multimate, tutorial PC-Tutor	4 ranuras	$2595
2S	BASIC	También tiene CPU Z80 con CP/M-80 y CP/M-86	$3495
2S, 1P	BASIC, CP/M-86	Ranuras, portátil (Spirit)	$2795 (Plus); $4295 (Plus XL); $3295 (Spirit
1S, 1P	WordStar, SuperCalc, CP/M-86	6 ranuras, puede usar otras CPU	$3995
1S	BASIC	Coge en una cartera, tiene panel táctil (similar en uso a un ratón), modem interno de 300 baudios, batería interna, sistema operativo propio y software incluido	$3995

Tabla D-1. Comparación de computadoras MS-DOS *(continuación)*

Nombre y modelo	CPU	RAM	Unidades de disco	Monitor
Hewlett-Packard 150	8088	256K	Dos de 3,5 pulgadas (270K cada una)	M 512×390
ITT XTRA	8088	128K	1	M 640×200
Leading Edge PC	8088	128K	2	M 640×200
Panasonic Senior Partner	8088	128K	1	M 640×200
Polo	80188	128K	2	C 640×200
Sanyo MBC 550	8088	128K	1 160K	M
Seattle Computer Gazelle II	8086	256K	Dos de 1,3M y 8 pulgadas	No incluido (usa terminal)
Seequa Chameleon Plus	8088	256K	2	M 640×200
Tandy TRS-80 Model 2000	80186	128K	2	Ninguno incluido
TeleVideo TS 1605	8088	128K	2	M 640×200
Texas Instruments Professional	8088	64K	1	M 640×200

Puertas	Software incluido	Otras características	Precio
2S, 1 GPIB	BASIC, Personal Applications Manager, tutor	Pantalla táctil	$3995
1S, 1P	BASIC	Ratón	$3500
1S, 1P	BASIC, Multiplan, procesador de texto	Reloj	$2895
1S, 1P	WordStar, Visicalc, PFS FIlei/Report/Graph	Portátil, incluye impresora térmica	$2495
2S	BASIC, Multiplan, PFS Write/Graph/File/Report	Incluye modem, impresora, CPU Z80 con CP/M	$1398
1P	Easy Writer, Wordstar, CalcStar	Puerta para juegos	$1398
2S, 1P	Editor	Bus S-100	$6995
1S, 1P	BASIC, Perfect Writer, Perfect Calc, Perfect Speller, Condor I, C-Term	Portátil, también tiene CPU Z80 y CP/M	$2895
1S, 1P	BASIC	4 ranuras	$2750
1S, 1P		1 ranura	$2995
1P		4 ranuras	$2195

E

FABRICANTES DE HARDWARE Y SOFTWARE

Muchos fabricantes producen computadoras MS-DOS. La tabla E-2 proporciona nombres y direcciones de los fabricantes de las computadoras mencionadas en el apéndice D.

Los capítulos 5 y 6 describen varios productos software para computadoras MS-DOS. La tabla E-1 proporciona las direcciones de los fabricantes que hacen dichos productos software.

Tabla E-1. Nombres y direcciones de los fabricantes de software

ABW Corporation P.O. Box M 1047 Ann Arbor, MI 48106	ATI Training Company 3770 Highland Avenue, Suite 201 Manhattan Beach, CA 90266
AmeriSoft 345 S. McDowell Blvd., Suite 410 Petaluma, CA 94952	Autodesk, Inc. 150 Shoreline Highway, #B20 Mill Valley, CA 94941
Anderson-Bell P.O. Box 191 Canon City, CO 81212	Aware Company P.O. Box 695 Gualala, CA 95445
Ann Arbor Software 407 N. Main Street Ann Arbor, MI 48104	Bellesoft, Inc. 2127 Bellevue Way SE Bellevue, WA 98031
Application Executive Corporation 600 Broadway, Suite 4C New York, NY 10012	BG Graphics 824 Stetson Avenue Kent, WA 98031
Ashton-Tate 10150 W. Jefferson Blvd. Culver City, CA 90230	Blue Chip Software 19824 Ventura Blvd., Suite 125 Woodland Hills, CA 91364

Tabla E-1. Nombres y direcciones de los fabricantes de software *(continuación)*

Bluebush, Inc.
P.O. Box 3585
Santa Clara, CA 95055

Bourbaki Inc.
P.O. Box 2867
Boise, ID 83701

Broderbund Software
1938 Fourth Street
San Rafael, CA 94901

Bruce and James
4500 Tuller Road
Dublin, OH 43017

Jim Button
P.O. Box 5786
Bellevue, WA 98006

C Ware Corporation
1607 New Brunswick
Sunnyvale, CA 94087

Carousel Microtools, Inc.
609 Kearney Street
El Cerrito, CA 94530

Cdex Corporation
5050 El Camino Real
Los Altos, CA 94022

Central Point Software, Inc.
P.O. Box 19730
Portland, OR 97219

Chang Laboratories, Inc.
5300 Stevens Creek Blvd., #200
San Jose, CA 95129

Comprehensive Software Support
2316 Artesia Blvd., Suite B
Redondo Beach, CA 90278

Computech Group Inc.
Main Line Industrial Park, Lee Blvd.
Frazer, PA 19355

Computer Advanced Ideas
1442A Walnut Street, Suite 341
Berkeley, CA 94709

Computer Aided Design
764 24th Avenue
San Francisco, CA 94121

Computer Innovations
10 Machanic Street, Suite J
Redbank, NJ 07701

Computer Shack, Inc.
P.O. Box 190
Etobicoke, Ontario, CANADA
M9C 4V3

Computing!
2519 Greenwich Street
San Francisco, CA 94123

CompuView Products, Inc.
1955 Pauline Blvd., Suite 200
Ann Arbor, MI 48103

Context Management Systems
23868 Hawthorne Blvd.
Torrance, CA 90505

Courseware, Inc.
10075 Carroll Canyon Road
San Diego, CA 92131

c-systems
P.O. Box 3253
Fullerton, CA 92634

Datasoft, Inc.
536 Valley Way
Milpitas, CA 95035

Dayflo Software
2500 Michelson Drive, Building 400
Irvine, CA 92715

Designer Software
3400 Montrose Blvd., Suite 718
Houston, TX 77006

Digital Marketing
2670 Cherry Lane
Walnut Creek, CA 94596

Digital Research
160 Central Avenue
Pacific Grove, CA 93950

Digital Signature
5453 S. Woodlawn
Chicago, IL 60615

Tabla E-1. Nombres y direcciones de los fabricantes de software *(continuación)*

Duncan-Atwell Computerized Technologies
924 Worth Avenue #B-2
Linden, NJ 07036

Earth Data Corporation
P.O. Box 13168
Richmond, VA 23225

Ecosoft Inc.
P.O. Box 68602
Indianapolis, IN 46268

Electronic Arts
2755 Campus Drive
San Mateo, CA 94403

Emerging Technology
2031 Broadway
Boulder, CO 80302

Enertronics Research, Inc.
150 N. Meramec, Suite 207
St. Louis, MO 63105

Ensign Software
7337 Northview
Boise, ID 83704

Fox & Geller, Inc.
604 Market Street
Elmwood Park, NJ 07407

FriendlySoft, Inc.
3609 Smith-Barry Road
Arlington, TX 76103

Graphic Software Systems
25117 SW Parkway
Wilsonville, OR 97070

Harvard Associates, Inc.
260 Beacon Street
Somerville, MA 02143

Headlands Press, Inc.
P.O. Box 862
Tiburon, CA 94920

Heritage Group
611 Anton Blvd., Suite 720
Costa Mesa, CA 92626

Heritage Software, Inc.
2130 S. Vermont Avenue
Los Angeles, CA 90007

IBM
P.O. Box 1328
Boca Raton, FL 33432

IMSI
633 Fifth Avenue
San Rafael, CA 94901

Infocom, Inc.
55 Wheeler Street
Cambridge, MA 02138

Information Unlimited Software, Inc.
2401 Marinship Way
Sausalito, CA 94965

Innovative Software
9300 W. 110th Street, Suite 380
Overland Park, KS 66210

International Software Alliance
1835 Mission Ridge Road
Santa Barbara, CA 93103

ITSoftware
P.O. Box 2392
Princeton, NJ 08540

Knoware
301 Vassar Street
Cambridge, MA 02139

Laboratory Microsystems, Inc.
4147 Beethoven Street
Los Angeles, CA 90066

Lifeboat Associates
1651 Third Avenue
New York, NY 10028

Lifetree Software, Inc.
411 Pacific Street, Suite 315
Monterey, CA 93940

Lightning Software
P.O. Box 11725
Palo Alto, CA 94306

Tabla E-1. Nombres y direcciones de los fabricantes de software *(continuación)*

Living Videotext, Inc.
1000 Elwell Court, Suite 232
Palo Alto, CA 94303

Logical Systems
1355 Sunrise Court
Los Altos, CA 94022

Logitech
165 University Avenue
Palo Alto, CA 94301

Lotus Corp.
55 Wheeler Street
Cambridge, MA 02138

LXI, Inc.
P.O. Box 3032
West Lafayette, IN 47906

Mansfield Software Group
Box 532
Storrs, CT 06268

Manx Software Systems
P.O. Box 55
Shrewsbury, NJ 07701

Mark of the Unicorn
222 Third Street
Cambridge, MA 02142

Mark Williams Company
1430 West Wrightwood
Chicago, IL 60614

Martin Marietta
P.O. Box 2392
Princeton, NJ 08540

Mathtech, Inc.
P.O. Box 2392
Princeton, NJ 08540

Mattel Electronics
5180 Rosecrans Avenue
Hawthorne, CA 90250

mbp Software
7700 Edgewater Drive, Suite 360
Oakland, CA 94621

Metasoft Corporation
6509 West Frye Road
Chandler, AZ 85224

Micro Data Base Systems, Inc.
P.O. Box 248
Lafayette, IN 47902

MicroBusiness Software, Inc.
Dover Road
Chichester, NH 03263

Microcomputer Taxsystems, Inc.
6203 Variel Avenue, Suite E
Woodland Hills, CA 91367

Micrografx
8526 Vista View
Dallas, TX 75243

MicroPro
33 San Pablo Avenue
San Rafael, CA 94903

MicroRIM
P.O. Box 585
Bellevue, WA 98004

Microsoft, Inc.
10700 Northup Way
Bellevue, WA 98004

Microstuf
1845 The Exchange, Suite 140
Atlanta, GA 30339

Microtaure Inc.
P.O. Box 6039, Station «J»
Ottowa, Ontario CANADA K2A1T1

MicroType
6531 Crown Blvd., # 3A
San José, CA 95120

MicroWare
P.O. Box 79
Kingston, MA 02364

MLI Microsystems
P.O. Box 825
Framingham, MA 01701

Mountain View Press, Inc.
P.O. Box 4656
Mountain View, CA 94040

MPR/Mathematica
P.O. Box 2393
Princeton, NJ 08540

Tabla E-1. Nombres y direcciones de los fabricantes de software *(continuación)*

Norell Data Systems
3400 Wilshire Blvd.
P.O. Box 70127
Los Angeles, CA 90010

Omega MicroWare, Inc.
222 S. Riverside Plaza
Chicago IL 60606

Omnisoft Corporation
9960 Owensmouth Avenue, Suite 32
Chatsworth, CA 91311

Orion Software
P.O. Box 2488
Auburn, AL 36831

Pacific Basin Graphics
1577 Ninth Avenue
San Francisco, CA 94122

PC TechniCorp
40 Grove Street
Wellesley, MA 02181

PCsoftware
9120 Gramercy Drive, Suite 416
San Diego, CA 92123

Peachtree Software
3445 Peachtree Road N.E.
Atlanta, GA 30326

Penguin Software
830 Fourth Avenue, Box 311
Geneva, IL 60134

Perfect Software
702 Harrison Street
Berkeley, CA 94710

Personal CAD Systems, Inc.
15425 Los Gatos Blvd.
Los Gatos, CA 95030

Peter Norton
2210 Wilshire Blvd., #186A
Santa Monica, CA 90403

PFS
422 Aldo
Santa Clara, CA 95050

Phoenix Software Associates, Ltd.
1 Knollwood Street
North Easton, MA 02356

Prentice-Hall, Inc.
200 Old Tappan Road
Old Tappan, NJ 07675

Professional Software, Inc.
51 Fremont Street
Needham, MA 02194

Programmers Shop
908-1 Providence Highway
Dedham, MA 02026

Proper Software
2000 Center Street, Suite 1024
Berkeley, CA 94704

Pyramid Data, Ltd.
P.O. Box 10116
Santa Ana, CA 92711

Quarterdeck Software
1918 Main Street, Suite 240
Santa Monica, CA 90405

Quest Research, Inc.
P.O. Box 2553
Huntsville, AL 35804

Quicksoft
219 First N. #224
Seattle, WA 98109

ReadiWare Systems Inc.
Box 107A
Ridgefield, CT 06877

Redding Group
609 Main Street
Ridgefield, CT 06877

Robert J. Brady Co.
Games Division
Bowie, MD 20715

Rocky Mountain Software Systems
1280-C Newell Avenue, Suite 147-J
Walnut Creek, CA 94596

Tabla E-1. Nombres y direcciones de los fabricantes de software *(continuación)*

RoseSoft, Inc.
4710 University Way, N.E., Suite 601
Seattle, WA 98105

RR Software
P.O. Box 1512
Madison, W1 53701

Ryan-McFarland
609 Deep Valley Drive
Rolling Hills Estates, CA 90274

Satellite Software International
288 West Center
Orem, UT 84057

Schuchardt Software Systems
515 Northgate Drive
San Rafael, CA 94903

Scitor Corporation
710 Lakeway, Suite 290
Sunnyvale, CA 94086

Sierra On-Line, Inc.
36757 Mudge Rance Road
Coarsegold, CA 93614

Soft Lab
P.O. Box 2186
2545 Parry Avenue North
Bremerton, WA 98310

Soft Warehouse
P.O. Box 11174
Honolulu, HI 96828

SoftCorp, Inc.
2340 State Road 580, Suite B244
Clearwater, FL 33575

SoftCraft
P.O. Box 9802
Austin, TX 78766

SofTool Systems
8972 E. Hampden Avenue, # 179

Software 128
363 Walden Street
Concord, MA 01742

Software Arts
27 Mica Lane
Wellesley, MA 02181

Software Link, Inc.
6700 23-B Roswell Road
Atlanta, GA 30328

Software Products International
10240 Sorrento Valley Road
San Diego, CA 92121

Software Technology for Computers
153 California Street
Newton, MA 02158

Softword Systems, Inc.
52 Oakland Avenue N.
East Hartford, CT 06108

Sorcim
2310 Lundy Avenue
San Jose, CA 95131

Spinnaker Software Corp.
215 First Street
Cambridge, MA 02142

Stoneware, Inc.
50 Belvedere Street
San Rafael, CA 94901

Structured Systems Group
5204 Claremont Avenue
Oakland, CA 94618

STSC, Inc.
2115 East Jefferson Street
Rockville, MD 20852

subLogic Corporation
713 Edgebrook Drive
Champaign, IL 61820

SuperSoft
P.O. Box 1628
Champaign, IL. 61820

Suttle Enterprises
29844 W. Chicago
Livonia, MI 48150

Symmetric Software
1805 Clemson Street
San Bernardino, CA 92407

Telecon Systems
1155 Meridian Avenue, Suite 218
San Jose, CA 95125

Tabla E-1. Nombres y direcciones de los fabricantes de software *(continuación)*

Time Arts, Inc.
4425 Cavedale Road
Glen Ellen, CA 95442

Trigram Systems
3 Bayard Road, #66
Pittsburgh, PA 15213

Trillian Computer
P.O. Box 481
Los Gatos, CA 95031

Ultragraphics Systems
1100 South Main
Racine, WI 53403

Vandata
17544 Midvale Avenue N., Suite 107
Seattle, WA 98133

VisiCorp
2895 Zanker Road
San Jose, CA 95134

Visual Age
642 N. Larchmont Blvd.
Los Angeles, CA 90004

Wadsworth Electronic Publishing Company
Statler Office Building, 20 Park Plaza
Boston, MA 02116

Washington Computer Services
3028 Silvern Lane
Bellingham, WA 98226

Waterloo Mycrosystems Inc.
175 Columbia Street West
Waterloo, Ontario, CANADA N2L 3B6

WATSOFT Products, Inc.
158 University Avenue
Waterloo, Ontario, CANADA N2L 3E9

Windmill Software, Inc.
2209 Leominster Drive
Burlington, Ontario, CANADA L7P 3W8

Word Associates
55 Sutter Street, #361
San Francisco, CA 94104

XOR Corporation
5421 Opportunity Court
Minnetonka, MN 55343

XyQuest Inc.
P.O. Box 372
Bedford, MA 01730

Tabla E-2. Nombres y direcciones de los fabricantes de hardware

ACT North America, Inc.
3375 Scott Blvd., Suite 336
Santa Clara, CA 95051

Bytec Corporation
8 Colonnade Road
Ottawa, Ontario, CANADA K2E 7M6

Columbia Data Products, Inc.
9150 Rumsey Road
Columbia, MD 21045

Compaq Computer Corporation
12330 Perry Road
Houston, TX 77070

Corona Data Systems
31324 Via Colinas, Suite 110
Westlake Village, CA 91362

Digital Equipment Corporation
2 Mt. Royal Avenue
Marlboro, MA 01752

Eagle Computer, Inc.
983 University Avenue
Los Gatos, CA 95030

Fujitsu Professional Microsystems
3320 Scott Blvd.
Santa Clara, CA 95051

Gavilan Computer Corporation
240 Hacienda Avenue
Campbell, CA 95008

Hewlett-Packard
974 E. Arques Avenue
Sunnyvale, CA 94086

International Businness Machines
Armonk, NY 10504

ITT
1515 West 14th Street
Tempe, AZ 85281

Leading Edge
225 Turnpike Street
Canton, MA 02021

Panasonic
1 Panasonic Way
Secaucus, NJ 07094

Polo Microsystems, Inc.
2570 El Camino Real
Mountain View, CA 94040

Sanyo
51 Joseph Street
Moonachie, NJ 07074

Seattle Computer Products
1114 Industry Drive
Seattle, WA 98188

Seequa Computer Corporation
8305 Telegraph Road
Odenton, MD 21113

Tandy Corporation/Radio Shack
1800 One Tandy Center
Fort Worth, TX 76102

TeleVideo
1170 Morse Avenue
Sunnyvale, CA 94086

Texas Instruments
P.O. Box 402430
Dallas, TX 75240

F

GLOSARIO

Archivar. Hacer copias de ficheros o discos. Ver también *copia de seguridad.*

Argumento. Un modificante que da más información a una orden. Por ejemplo, la orden TYPE toma como argumento el nombre del fichero que va a ser visualizado.

Argumento genérico. Argumento en la sintaxis de una orden que puede tomar muchos valores. Un argumento genérico se usa exclusivamente a título ilustrativo.

ASCII. Código estándar utilizado para emparejar números con caracteres. Puesto que las computadoras sólo piensan en términos de números, cada carácter tiene que tener un número asociado con él (por ejemplo, A es el 65 y B el 66). ASCII son las siglas del grupo que desarrolló el estándar, el American Standard Code for Information Interchange.

Asterisco. Símbolo (*) usado en un especificador de fichero como un carácter comodín. El asterisco puede sustituir a cualquier número de caracteres en la posición en la que aparece. La interrogación es otro carácter comodín.

AUTOEXEC.BAT. Fichero batch que se ejecuta automáticamente cuando se hace la carga inicial de la computadora.

Auto-marcado. Capacidad de un modem para llamar a otras computadoras sin que el usuario marque el teléfono.

Banco. Conjunto de 64K palabras de memoria RAM. La memoria adicional se suele añadir a una computadora en forma de bancos.

Base de datos. Colección estructurada de informaciones relacionadas entre sí, almacenada en disco.

Baudio. Medida de la rapidez con que se transfieren los datos entre una computadora y un dispositivo. Un baudio equivale a 1 bit por segundo; 300 baudios son aproximadamente 30 caracteres por segundo.

Bit. La unidad de memoria más pequeña; consta de un conmutador «on/off». Un bit es un octavo de byte.

Bits de datos. Protocolo que indica cuantos bits de información son datos en el flujo de comunicación.

Bits de parada. Protocolo que indica cuantos bits de información siguen a los bits de datos en el flujo de comunicaciones.

Byte. Medida estándar de memoria, tanto para RAM como para memoria de masa. Un byte es un grupo de 8 bits que representa a un número o a un carácter. Un byte puede tener uno de 256 valores y puede contener un carácter alfanumérico.

Cadena. Secuencia de caracteres. Las cadenas se usan en algunas órdenes del MS-DOS para dar a la orden un trozo de texto.

Caída. Parada completa. Si se produce, es necesario realizar de nuevo la carga inicial.

Cálculo de «qué-pasa-si». Cálculo en el que se puede cambiar alguna de las suposiciones. Después de calcular la solución de una ecuación, se puede cambiar el valor de una de las variables y volver a hacer el cálculo.

Camino. Descripción de cómo un subdirectorio se enlaza con otro directorio o subdirectorio.

Campo. Subsección de un registro en una base de datos, que contiene un solo tipo de información. Ver también *registro.*

Campo clave. Campo acerca del cual un sistema de gestión de base de datos guarda información especial. Encontrar valores en campos clave es mucho más rápido que encontrarlos en campos normales.

Cápsula (Shell). Programa que actúa como intermediario entre el usuario y el sistema operativo.

Carácter comodín. Símbolo usado en un especificador de fichero para igualar con cualquier carácter en la posición donde aparece. Ver también *interrogación* y *asterisco.*

Carga inicial. Carga el sistema operativo en memoria RAM. El sistema operativo se carga cada vez que se pone en marcha la computadora o cuando se pulsa la tecla RESET.

Cargar. Transferir un programa desde la memoria de masa a la RAM.

Celdilla. Posición específica en la cuadrícula de una hoja cálculo. Una celdilla contiene información para ser procesada por el programa.

Chip de soporte. Chip que ayuda a la CPU para que haga menor procesamiento, consiguiendo así que la computadora sea más rápida.

Cifrado. Conversión de un fichero legible a una forma ilegible utilizando un código. El cifrado de un fichero evita que las personas no autorizadas puedan comprender el contenido del fichero.

Cola. Lista de ficheros esperando para ser escritos por impresora.

Compilador. Programa que convierte texto escrito en un lenguaje de programación, en instrucciones en lenguaje máquina. El programa resultante puede ejecutarse igual que un programa de aplicación. Ver también *intérprete.*

Computadora de funcionamiento similar al IMB PC. Computadora que ejecuta la mayor parte del software ejecutable sobre el IBM PC.

Concatenar. Conectar dos ficheros, uno al final del otro.

Consola. Conjunto de teclado y pantalla. La consola se utiliza para dar órdenes al MS-DOS y ver los resultados.

Copia de seguridad. Copia de un fichero o disco que puede utilizarse en caso de que el original se estropee. La copia de seguridad de datos y programas es necesaria para evitar su pérdida, en caso de que se dañen accidentalmente.

Cursor. Caja o subrayado en la pantalla que la computadora usa para indicar dónde aparecerá el siguiente carácter que se teclee.

DBMS relacional. Sistema de gestión de base de datos que puede relacionar la información de un fichero con la información en otros ficheros.

Depuración. Corrección de los errores de un programa.

Descifrado. Conversión de un fichero codificado en otro legible. Ver *cifrado.*

Desplazamiento (Scroll). Movimiento de un extremo a otro de un fichero texto, línea por línea.

Dirección. Un número que representa una posición exacta en RAM.

Directorio. Area de cada disco que contiene una lista de todos los ficheros contenidos en ese disco.

Directorio de forma de árbol o directorio arborescente. Sistema para organizar los ficheros en un disco. Un directorio en forma de árbol tiene subdirectorios que son ramificaciones del directorio raíz.

Directorio raíz. Directorio base de un árbol. Cuando se hace la carga inicial del MS-DOS el directorio raíz es el subdirectorio por defecto.

Disco. Medio de almacenamiento hecho de metal o plástico recubierto de material magnético. El disco gira a gran velocidad en una unidad de disco y la información se lee con una cabeza magnética (similar a la cabeza de un magnetofón). Ver también *disquete* y *disco rígido.*

Disco flexible. Ver *disquete.*

Disco por defecto. Disco en el que el MS-DOS busca los programas y los datos a menos que se le indique que los busque en otro sito.

Disco del sistema. Disco que se usa para la carga inicial del MS-DOS. El disquete de distribución para un programa de aplicación también se puede llamar disquete del sistema.

Disco RAM. Método que permite al MS-DOS tratar parte de la RAM como una unidad de disco.

Disco rígido o duro. Disco rígido revestido magnéticamente. Los discos rígidos contienen más información que los discos flexibles y el acceso a los datos se realiza más rápidamente. Son más caros que la mayoría de las otras formas de memoria masiva.

Disco rígido extraíble. Disco rígido que tiene un medio magnético extraíble.

Disco rígido fijo. Disco rígido que no puede extraerse de la unidad de disco.

Disquete. Disco de bajo coste hecho de material flexible (también llamado disco flexible). Los disquetes tienen usualmente 5 1/4 pulgadas, aunque algunas computadoras usan disquetes de 8 pulgadas; se puede extraer de la unidad de disco y almacenarlos por separado.

Disquete de distribución. Disquete que se recibe cuando se compra software. El disquete de distribución contiene el programa de aplicación.

Dispositivo. Componente hardware usada para entrada o salida.

Doble punto. Símbolo (..) utilizado en un camino para representar el padre del subdirectorio por defecto.

Eco. Repetición de la información en la pantalla. Por ejemplo, la orden ECHO visualiza sus argumentos en la pantalla.

Editar. Cambiar un fichero texto utilizando un programa.

Editor de líneas. Programa editor de texto que sólo visualiza una línea del fichero mientras se está editando. Para comparar, ver *editor orientado a pantalla.*

Editor orientado a pantalla. Programa editor de texto que utiliza la mayoría de las líneas de la pantalla para visualizar un fichero mientras se edita. Para comparar, ver *editor de líneas.*

Editor de texto. Programa que permite crear y cambiar los contenidos de un fichero texto.

Emulador de terminal. Programa que imita las acciones de un terminal.

Entorno. Area de la RAM que el MS-DOS usa para guardar variables. Se puede modificar y consultar con la orden SET. Por ejemplo, la cadena de petición de orden (prompt) se guarda como una variable de entorno.

Entrada. Información usada por un programa. Introducir los datos de entrada significa dar información a un programa cuando la solicita.

Entrada redirigida. Datos que normalmente se introducen por el teclado, pero que en lugar de eso se obtienen de un fichero. Un programa que usa la entrada redirigida lee sus datos de entrada del fichero en lugar de del teclado.

Error de fuera de tiempo. Error que ocurre cuando el MS-DOS intenta enviar información a una puerta, pero no recibe confirmación de que la puerta está dispuesta.

Especificador de fichero. Identificador completo de un fichero. El especificador completo incluye el nombre de fichero y la extensión. Puede incluir también el nombre del disco y el nombre del camino.

Especifichero. Ver *especificador de fichero.*

Estructura. Descripción de la organización global de una base de datos.

Extensión. La segunda parte de un especificador de fichero que aparece a la derecha del punto. La extensión describe usualmente el tipo del fichero y puede tener hasta tres caracteres. Se llama también tipo del fichero.

Fichero. Colección de informaciones que se almacena en disco y a la que se accede usando un único especificador de fichero.

Fichero binario. Fichero que contiene caracteres especiales que no son de tipo texto. Estos ficheros son usualmente programas y bases de datos.

Fichero contiguo. Fichero cuyos bloques de datos están situados uno a continuación de otro en el disco.

Fichero oculto. Fichero que no aparece en el listado del directorio del disco. Los ficheros ocultos tienen un indicador especial en su entrada del directorio que impide que las órdenes del MS-DOS actúen sobre ellos.

Fichero de tratamiento por lotes (Fichero «batch»). Fichero texto que contiene órdenes del MS-DOS. El MS-DOS ejecuta las órdenes de un fichero «batch» como si fueran un programa. Los ficheros «batch» siempre llevan la extensión BAT.

Ficheros del sistema. Ficheros usados exclusivamente por el MS-DOS.

Filtro. Programa que toma como entrada la salida de otro programa.

Formateado. Proceso que consiste en colocar marcas magnéticas en un disco para que pueda ser leído y grabado por una unidad de disco. La orden FORMAT se utiliza para realizar esta operación antes de usar un disco nuevo.

Fragmentación. División en partes. Un fichero fragmentado es aquel cuyas partes están en distintos sitios del disco. El MS-DOS accede a un disco con muchos ficheros fragmentados de forma más lenta que cuando accede a ficheros contiguos (no fragmentados).

Fundición. Tipos de letra. Algunos tipos comunes son redondilla, cursiva y negrita.

Hardware. Parte física de una computadora, tal como la CPU, las unidades de disco o el monitor.

Hoja de cálculo. Programa que calcula números sobre un cuadriculado similar a una hoja de contabilidad. Cuando una parte de la hoja se cambia, otras partes, que están relacionadas con la parte cambiada, se actualizan automáticamente.

Impresora de matriz de puntos. Impresora que forma los caracteres sobre el papel presionando una serie de puntas contra una cinta entintada.

Indicadores del sistema. Características del MS-DOS que pueden ser activadas con las órdenes del MS-DOS (tales como VERIFY).

Inteligencia artificial (AI). El campo de la informática que se ocupa de desarrrollar para las computadoras características más similares a las humanas, tales como razonamiento deductivo, visión y aprendizaje.

Interactivo. Capaz de realizar preguntas y dar respuestas. Un programa interactivo pide información mientras se ejecuta.

Intérprete. Programa que lee y ejecuta otro programa paso a paso. Los programas interpretados no pueden ejecutarse desde el MS-DOS, sino que tienen que ejecutarse desde dentro del intérprete.

Intérprete de órdenes. Parte del MS-DOS que lee órdenes, busca los programas de órdenes y comienza los programas de órdenes.

Interrogación. Símbolo (?) utilizado en un especificador de fichero como carácter comodín. La interrogación puede sustituir a cualquier carácter en la posición en que aparece. El asterisco es otro carácter comodín.

I/O (E/S). Entrada y salida.

K (Kilobyte). 1.024 bytes. La capacidad de la memoria y del disco se mide usualmente en K bytes (así, 64 K son 65.536 bytes).

Lenguaje. Programa que se utiliza para escribir otros programas.

Lenguaje ensamblador. Lenguaje de programación usado para controlar la CPU directamente.

Maestro. Disco original en el que se distribuye el software.

Marca fin de fichero. Carácter que especifica el final de un fichero texto.

Megabyte. Unidad de medida igual a 1.024 kilobytes ó 1.048.576 bytes.

Mejora. Adición de hardware a la computadora para incrementar su versatilidad o capacidad.

Memoria. Partes de la computadora o dispositivos asociados que pueden almacenar información o programas. Ver también *memoria de acceso aleatorio* y *disco.*

Memoria de acceso aleatorio (RAM). Chips de memoria que contienen la información que usa la comptuadora.

Memoria masiva. Dispositivos que guardan información después de desconectar la computadora. Las memorias masivas más comunes son los discos y las cintas.

Memoria de sólo lectura (ROM). Memoria que sólo puede ser leída pero no cambiada. La ROM se usa para almacenar programas pequeños que no cambian nunca.

Mensaje de error. Mensaje que se imprime en la pantalla que indica que un programa no puede terminar normalmente.

Menú. Lista de opciones dada por un programa en la pantalla. Una opción se selecciona pulsando una tecla o escribiendo una palabra. Debido a que los menús muestran todas las alternativas son más fáciles de usar que las peticiones de órdenes.

Modem. Dispositivo que permite comunicarse con otras computadoras a través de líneas telefónicas normales.

Monitor. Pantalla de la computadora que se utiliza para visualizar texto y gráficos.

Monocromo. De un color. Los monitores monocromos son la mayor parte de las veces negro y verde, negro y blanco o negro y ámbar.

Muesca de protección de escritura. Muesca situada en la parte superior derecha de un disco de 5 1/4 pulgadas. Cuando la muesca está tapada, no se puede grabar ninguna información sobre el disco.

Multitarea. Capacidad para ejecutar en una computadora varios programas al mismo tiempo.

Multiusuario. Capacidad para que mucha gente ejecute programas simultáneamente en una computadora.

Nivel de error. Número que indica si un programa terminó correctamente o qué es lo que evitó la terminación correcta del programa. La orden IF permite consultar el nivel de error.

Nombre de camino. Nombre que el MS-DOS utiliza para describir la conexión entre un subdirectorio y la raíz.

Nombre de camino absoluto. Un nombre de camino que muestra cómo un subdirectorio se enlaza con el directorio raíz.

Nombre de camino relativo. Nombre de camino que muestra cómo un subdirectorio se enlaza con el subdirectorio por defecto.

Nombre de fichero. La parte del especificador de fichero situada a la izquierda del punto. Un nombre de fichero puede tener hasta 8 caracteres.

Orden. Instrucción dada al MS-DOS. Ver también *orden interna* y *orden externa.*

Orden de búsqueda. El orden en el que el MS-DOS busca los ficheros de órdenes. El orden de búsqueda se establece y se visualiza con la orden PATH.

Orden externa. Orden que se almacena en disco en vez de en RAM. Cada vez que se introduce una orden externa el MS-DOS tiene que leerla del disco para ejecutarla. En el MS-DOS las órdenes externas tienen la extensión COM o EXE. Ver también *orden interna.*

Orden interna. Orden que se guarda en RAM una vez que se ha realizado la carga inicial del MS-DOS. A diferencia de las órdenes externas, el MS-DOS no necesita leer del disco las órdenes internas cada vez que se utilizan.

Ordenes de formateado. En el procesamiento de textos, las órdenes que se utilizan para cambiar el aspecto del texto en la páginas impresa.

Ordenes de tratamiento por lotes (Ordenes «batch»). Ordenes del MS-DOS que se usan sólo en ficheros de tratamiento por lotes. Estas órdenes son ECHO, FOR, GOTO, IF, PAUSE, REM, y SHIFT.

Paridad. Protocolo usado en la comunicación de dos dispositivos hardware. La paridad se usa para incrementar la probabilidad de que cada byte enviado sea correcto.

PC-DOS. Variación del sistema operativo MS-DOS usada en el IBM PC.

Programa residente. Programa que permanece en RAM después de ser cargado.

Periférico. Elemento hardware añadido a una computadora. Algunos periféricos comunes son las impresoras y los modems.

Petición de orden (Prompt). Mensaje que se imprime cuando el MS-DOS o un programa está esperando a que se escriba una orden. La petición de orden por defecto del MS-DOS es A >.

Pista. Anillo concéntrico en un disco, similar a un surco en un disco de música.

Pistas de carga inicial. Area de un disco que guarda información sobre cómo la computadora debe cargar el sistema operativo.

Plantilla. Copia genérica de una aplicación en la cual la información puede ser colocada igual que en un impreso en el que hay que rellenar los espacios en blanco.

Por defecto. Algo que se asume a no ser que se especifique otra cosa.

Procesador de texto. Programa que permite editar y formatear texto.

Procesamiento concurrente. Capacidad para ejecutar dos programas al mismo tiempo.

Programa de aplicación. Un programa que realiza una tarea. Procesamiento de texto o gestión de bases de datos son ejemplos de programas de aplicaciones.

Programa de utilidad. Programa especializado que ayuda a trabajar con otros programas (por ejemplo, un programa depurador). Las utilidades son como distintas herramientas que forman un juego de herramientas.

Protrección contra copia. Procedimiento que algunos fabricantes de software utilizan para evitar que los usuarios copien el disquete de programas. Impide las órdenes de copia de disquete del MS-DOS.

Protección de escritura. Procedimiento que evita que se escriba sobre un disco. Esto se hace usualmente tapando la muesca de protección de escritura de un disquete.

Protocolo. Método estándar utilizado por los dispositivos hardware para comunicarse.

Protocolo de comunicaciones. Medio por el cual dos dispositivos hardware comunican entre sí.

Puerta (port). Enchufe que permite conectar la computadora a un dispositivo tal como una impresora o un modem.

Puerta (port) paralelo. Puerta de impresora que usa un protocolo que envía y recibe información al mismo tiempo. Ver también *puerta serie.*

Puerta (port) serie. Puerta que usa un protocolo para enviar y recibir información en momentos distintos. Ver también *puerta paralelo.*

Ranura de expansión. Clavija en la que se pueden añadir tarjetas de circuitos. Algunas computadoras tienen ranuras de expansión que permiten añadir RAM y hardware especial (tal como capacidades gráficas adicionales).

Ratón. Dispositivo hardware que sirve de mando para mover el cursor por la pantalla.

Registro. Colección de informaciones que constituyen una entidad en una base de datos. Un registro está formado por campos.

Rótulo de disco. Ver *rótulo de volumen.*

Rótulo de volumen. Area de un disco que contiene el nombre del disco.

Salida. Información generada por un programa. Generación de información. También se denomina escritura.

Salida redirigida. Caracteres que normalmente se visualizan por pantalla pero que en vez de esto se almacenan en un fichero. Un programa que usa la salida redirigida escribe caracteres en un fichero en lugar de visualizarlos en la pantalla.

Sector. Parte de una pista de un disco. Hay un número fijo de sectores por pista. Ver también *pista.*

Sector en mal estado. Sector de un disco, que el MS-DOS no puede leer. Usualmente se debe a un error físico en el disco, tal como una huella dactilar.

Signo de porcentaje. Símbolo (%) utilizado en ficheros batch para indicar un argumento reemplazable.

Sintaxis. El orden estándar en el que hay que dar los argumentos de una orden.

Sistema de gestión de base de datos (DBMS). Programa que permite introducir u obtener información en una base de datos.

Sistema operativo. Programa que controla la computadora y que permite ejecutar otros programas.

Software. Programas que realizan tareas tales como procesamiento de texto y contabilidad.

Software compartido. Software disponible gratuitamente y que puede ser traspasado a otras personas. Los autores de estos programas compartidos piden un donativo a la gente que los usa.

Software de dominio público. Software que no está registrado y que es gratuito.

Software incluido. Software que está incluido en el precio de compra de una computadora. Suele ser software de negocios, tal como procesadores de texto y comunicaciones.

Software integrado. Paquete de software que combina muchos programas (tales como hoja de cálculo, gráficos y gestión de base de datos) en un paquete que puede usar la información generada por cada uno de sus programas componentes.

Subdirectorio. Directorio que jerárquicamente está por debajo de otro directorio (ver también *directorio en forma de árbol*). La versión 2 del MS-DOS permite subdividir el directorio del disco en subdirectorios.

Subdirectorio hijo. Subdirectorio que está debajo de otro subdirectorio en un directorio arborescente. Ver también *subdirectorio padre.*

Subdirectorio padre. Subdirectorio que jerárquicamente está por encima de otro subdirectorio en un directorio arborescente. Ver también *subdirectorio hijo.*

Subdirectorio por defecto. Subdirectorio en el que el MS-DOS busca los programas y los datos a menos que se le indique que los busque en otro sitio.

Subprograma. Parte pequeña de un programa más largo. Los subprogramas tienen funciones individuales que son usadas por otros programas.

Sustitución de argumentos. El método que permite sustituir argumentos en las órdenes cuando se ejecuta un fichero batch.

Tabla de asignación de ficheros. Area de cada disco que el MS-DOS usa para determinar qué datos contenidos en el disco pertenecen a cada fichero.

Tarjeta adaptadora. Una tarjeta de expansión que permite añadir a su computadora monitores monocromáticos o de color (generalmente se usa sólo con el IBM PC y similares).

Tarjeta de circuito. Tarjeta con chips y conexiones.

Tecla Break. Tecla que hace parar a un programa que se está ejecuntado (en el IBM PC, es la tecla CONTROL-SCROLL LOCK).

Tecla programable. Tecla del teclado cuya función puede ser cambiada. Por ejemplo, se puede cambiar una tecla programable para que escriba la palabra PRINT cada vez que se pulse.

Tecla RESET. Tecla o conmutador en la computadora que hace que el hardware se comporte como si la computadora se apagara y se encendiera de nuevo.

Tipo de fichero. Ver *extensión.*

Trazador de gráficos (Plotter). Dispositivo, similar a una impresora, que dibuja imágenes gráficas con una pluma.

Tubo. Conexión entre dos programas en la cual la salida del primer programa se usa como entrada del segundo programa.

Unidad central de procesamiento (CPU). Cerebro primario de una computadora. La CPU es el chip que procesa los datos y ejecuta las instrucciones de un programa.

Unidad de cinta magnética. Unidad de cinta utilizada para copiar datos de los discos y en los discos.

Unidad de disco. Dispositivo que permite leer y escribir en un disco. La unidad consta de un motor que hace girar el disco y una cabeza magnética móvil que lee y escribe información.

Ventana temporal. «Minipantalla» que aparece temporalmente en la pantalla. Usualmente contiene menús de opciones.

Versión. Número asignado a la revisión actual de un programa.

MARCAS REGISTRADAS

Los nombres en cursiva son productos registrados de las compañías correspondientes, con las marcas registradas indicadas con un®.

1-2-3	Lotus Development Corp.
I Dir	Bourbaki, Inc.
87 BASIC	MicroWare
ABSTAT	Anderson-Bell
*APL*Plus®/PC*	STSC
Apple®	Apple Computer, Inc.
ASCOM	Dynamic Microprocessor Associates, Inc.
Atari	Warner Communications
AutoCAD	Autodesk, Inc.
Benchmark®	Metasoft
C86	Computer Innovations, Inc.
CADPLAN	Personal CAD Systems
CalcStar	MicroPro International Corp.
CBASIC-86	Digital Research
CBASIC Compiler	Digital Research
Commodore	Commodore Business Machines, Inc.
Context MBA	Context Management Systems
CP/M®	Digital Research
Crosstalk XVI	Microstuf, Inc.
C-window	C-Systems
Data Design	Insoft
dBASE II	Ashton-Tate
Delta Drawing	Spinnaker Software Corp.
Desktop Accountant	Rocky Mountain Software Systems
Disk Mechanic	MLI Microsystems
EasyWriter	Information Unlimited Software
EnerGraphics	EnerTronics Research, Inc.
FriendlyWare	FriendlySoft, Inc.
Global Thermonuclear War	Omnisoft Corp.
Hyperion	Bytec-Comterm, Inc.
IBM®	International Business Machines, Inc.
IBM®PC/XT	International Business Machines, Inc.
InfoScope	Microstuf, Inc.
InfoStar	MicroPro International Corp.
Janus/Ada®	RR Software
KnowledgeMan	Micro Data Base Systems
Micro 16	Fujitsu
MicroStat®	ECOsoft
Microsoft® Word	Microsoft Corp.
Millionaire	Blue Chip Software
Mousekattack	Sierra On-Line, Inc.
MS-DOS®	Microsoft Corp.
Multiplan	Microsoft Corp.
Number Cruncher	Pyramid Data
PC Logo	Harvard Associates, Inc.
PC/FORTH	Laboratory Microsystems, Inc.
PeachText 5000	Peachtree Software
Perfect Filer	Perfect Software
Perfect Writer	Perfect Software
PFS:® File	Software Publishing Corp.
PFS:® Graph	Software Publishing Corp.
PFS:® Report	Software Publishing Corp.
PFS:® Write	Software Publishing Corp.
Planetfall	Infocom
Project Scheduler	Scitor Corp.
ProKey	RoseSoft, Inc.
Radio Shack® Model 2000	Radio Shack, A Division of the Tandy Corp.
Rainbow 100	DEC
ReadiWriter	ReadiWare Systems
RealWorld	MicroBusiness Software
Savior	Omega MicroWare, Inc.
Starcross	Infocom
SuperCalc® 3	Sorcim Corp.
SuperWriter	Sorcim Corp.
Symphony	Lotus® Development Corp.
Textra	Ann Arbor Software
The Final Word	Mark of the Unicorn
Think Tank	Living Videotext, Inc.
T.I.M.®	Innovative Software
TK!Solver	Software Arts, Inc.
TRS-80	Radio Shack, A Division of the Tandy Corp.
UNIX	AT&T Technologies
VisiCalc®	VisiCorp
VisiWord	VisiCorp
Volkswriter®	Lifetree Software
WordPlus	Professional Software, Inc.
WordStar®	MicroPro International Corp.
WordVision	Bruce & James Program Publishers, Inc.
Zork®	Infocom

INDICE

Ada, 251
APL, 252
Argumentos, 21-23
genéricos, 49-50

BASIC, 247, 248-249

C, 249, 250
Caminos:
definición, 160
órdenes para mantenimiento, 120-130
Caracteres comodín:
uso con COPY, 32-34
uso con DIR, 75-76
uso con ERASE, 53
Caracteres para redirigir I/O, 175-176
COBOL, 250
Computadoras:
que ejecutan MS-DOS, 284-289
líneas de guía para comprar, 281
Comunicación serie frente a comunicación paralelo, 5
Copias de seguridad:
de discos del sistema, 24-27
eligiendo un método para, 188-190
con la orden DISKCOPY, 91-94
realización de, 190-191
Corte de energía eléctrica, 186

Directorios, 7, 74-78
(*Véase también* Directorios en forma de árbol)
Directorios en forma de árbol:
creación, 162-164
definición, 160-161
Disco del sistema, 16, 26-28
Discos:
comprobación del estado de los, 84-89
copia de seguridad, 24-27, 91-94, 188-190
cuidado de los, 187
errores, 186
formateado, 23-24
función de los, 6-8
en mal estado, 186, 196
órdenes para mantenimiento, 73-98
protección de escritura, 27-28
recuperación con los programas de utilidades, 197
Discos fijos (*véase* Discos rígidos)
Discos rígidos:
capacidad de almacenamiento de los, 8
copias de seguridad de los, 189-190
cuidado de los, 187
particiones, 96
Diseño asistido por computadora (CAD), 237
Dispositivos, 4-5
copiando en y de, 59
Disquetes:
capacidad de almacenamiento de los, 8
definición, 6
revistas en, 261
Distribuidores:
asistencia proporcionada por los, 202, 281
frente a compañías de venta por correo, 203
software, 197-198
Documentación, 194, 195, 202

Editores de texto, 28, 65-66, 208, 212-213
Entrada/salida (I/O), 4
redirigida, 174-178

Fichero AUTOEXEC. BAT, 40-41
Ficheros:
batch, 35-41
contiguos frente a segmentados, 86-87
copias de seguridad, 189
creación de, 28
definición, 6
grupos de, 32-34
nombres de, 29-31
ordenación, 153-155
órdenes para mantenimiento, 52-66
órdenes para salida de, 66-73
organización con subdirectorios, 160-174
restauración, 89-91
Ficheros batch, 10, 35-42
ejemplo de uso, 41-42
enlace de, 41
nombres para, 37
órdenes usadas en, 130-150
uso de argumentos en, 37-40
Ficheros del sistema, 25

FORTH, 251
FORTRAN, 250

Hardware:
- anuncios de, 261
- comunicación entre, 3-5
- definición, 1-2
- estableciendo modos de comunicación, 103-108
- factores que afectan al comportamiento del, 279-280
- fallo del, 186
- líneas de guía para comprar, 280-284

Historia MS-DOS, 12-14

IBM PC:
- compatibilidad con, 277-278
- ejecución de software multiusuario sobre, 243
- libros sobre, 263-264
- órdenes externas específicas del, 26-27
- parando un programa en, 52
- programas de copia de discos para, 191
- programas de recuperación de discos para, 197
- revistas sobre, 259-260
- software para, 201
- software de dominio público para, 204

Impresoras, 68-69
- dirigiendo la salida de las, 107
- estableciendo modos de comunicación, 103-104
- matriz de puntos, 230

Lenguaje máquina, 9, 247
Lenguajes de programación, 246, 253
- BASIC, 248-249
- C, 249-250
- COBOL, 250-251
- FORTRAN, 250
- Pascal, 250

Libros, computadora, 263, 265
Líneas de órdenes
- edición de, 181-183

LISP, 252
Logo, 250

Memoria de acceso aleatorio, 3-4
- cantidad utilizada por diferentes versiones, 273
- MS-DOS en, 11-12

Mensajes de error, 11, 191-195
Microprocesadores:
- 8086, 12-13
- 8088, 12

Modems, 238
Multitarea, 14, 243

Nombres de caminos, 165-168, 173-174
Nombres de dispositivos, 34-35
Nombres de ficheros, 29-31

Orden ASSIGN, 114-116
Orden BACKUP, 97
Orden BREAK, 112-113
Orden CD, 122
Orden CLS, 155
Orden COMMAND, 147-150
Orden COMP, 63-65
Orden COPY, 24-25, 57-62
- argumentos para la, 60-61
- sintaxis de la, 62
- usos comunes de la, 57-60

Orden CTTY, 116-117
Orden CHDIR, 122-124, 162-164
Orden CHKDSK, 84-89, 196
Orden DATE, 16-17, 99-100
Orden DEBUG, 158
Orden DEL, 52-53
Orden DIR, 44, 74-78, 168-169
Orden DISKCOMP, 94-96
Orden DISKCOPY, 91-94
Orden ECHO, 132-133
Orden EDLIN, 28, 65-66
Orden ERASE, 52-54
Orden EXE2BIN, 158
Orden FDISK, 96
Orden FIND, 150-153
Orden FOR, 139-142
Orden FORMAT, 23-25, 78-82
Orden GOTO, 142-144
Orden GRAPHICS, 155-156
Orden IF, 135-139
Orden LINK, 156-157
Orden MD, 120-121
Orden MKDIR, 120-122, 162-164
Orden MODE, 103-108
Orden MORE, 72-73
Orden PATH, 128-131, 172-173
Orden PAUSE, 133-135
Orden PRINT, 43, 68-72
Orden PROMPT, 110-112
Orden RD, 124-125
Orden RECOVER, 89-91, 196
Orden REM, 145-146
Orden REM, 54-55
Orden RENAME, 54-57
Orden RESTORE, 97-98
Orden RMDIR, 124-126, 170-172
Orden SET, 117-119

Orden SHIFT, 146-147
Orden SORT, 153-155
Orden SYS, 82-84
Orden TIME, 17-18, 100-101
Orden TREE, 126-128, 169-170
Orden TYPE, 42-43, 67-68
Orden VER, 108-109
Orden VERIFY, 101-102
Orden VOL, 109-110
Ordenes:
  añadidas a la versión, 2, 274-275
  argumentos en, 21-23
  conectadas en tubo, 179-181
  diferentes entre versiones, 46, 271
  ejecución, 9-12
  en ficheros batch, 35-41, 130-150
  internas frente a externas, 9-11
  mantenimiento de caminos, 120-130
  mantenimiento de discos, 73-98
  mantenimiento de ficheros, 52-66
  para los indicadores del sistema, 98-119
  parada, 51-52
  programación, 156-158
  salida de ficheros, 66-73
  secuenciales, 66
  usadas con ficheros, 28-34
  uso de nombres de dispositivos en, 34-35
Ordenes externas, 10-11
  frecuentes, 26-27
  y la unidad por defecto, 19-21
Ordenes internas, 9-10

Pantallas:
  borrado, 155
  estableciendo el modo de adaptado, 104-105
Paquetes de software integrado, 224-225
Pascal, 250
Peticiones de orden:
  cambio, 110-112
  definición, 9
  MS-DOS, 18-19
Pilot, 252
Programación:
  ayudas, 255
  órdenes usadas para, 156-158
Programas de aplicaciones, 205-246
  cápsula, 242-243
  comunicaciones, 237-238
  contabilidad, 226-228
  educativos, 240-241
  ejecución, 5-6
  gestión de base de datos, 217, 224
  gráficos, 230-237
  hoja de cálculo, 216
  integrados, 225-226
  juegos, 243-245
  mensajes de error en, 192-195
  planificación de proyectos, 229-230
  procesamiento de texto, 205-212
  realización de copias de seguridad de, 190-191
Programas cápsula, MS-DOS, 242-243
Programas de comunicaciones, 237-238
Programas de contabilidad, 226
Programas para la declaración de impuestos, 255
Programas educativos, 239-241
Programas estadísticos, 253
Programas gráficos, 230-237
  para dibujo, 236
  para negocios, 230-235
Programa de hoja de cálculo, 215-216
Programas de juegos, 243-246
Programas de planificación de proyectos, 228-229
Programas para potenciar el teclado, 253
Programas para procesamiento de texto, 205-212
  criterios para elegir, 207-209
  fundamentos de, 206-208
  visión general de los disponibles, 209-213
Programas de productividad personal, 254-255
Programas de resolución de fórmulas, 254
Programas de seguridad de los datos, 253
Programas que verifican la ortografía, 208-209, 213-214
Puertas, 4-5
  estableciendo modos de comunicación, 105-107
Puesta en marcha, 16-18

RAM (*véase* Memoria de acceso aleatorio)
Revistas, computadora, 257-262
  generales, 260-261
  sobre el IBM PC, 259-260
Rótulo de disco (*véase* Rótulo de volumen)
Rótulo de volumen, 75, 80, 109-110

Sintaxis, orden, 49-51
Sistema operativo CP/M, 13-14
Sistema operativo de disco (DOS), 2
Sistema operativo PC-DOS, 2, 283
Sistema operativo UNIX, 14
Sistemas de gestión de base de datos (DBMS), 217-223
  criterios para elegir, 221-222
  fundamentos de, 217-220
  visión general de los disponibles, 222-224
Sistemas operativos multiusuario, 14, 243
Software (*véase* Programas de aplicación)

anuncios de, 261
de dominio público, 204-205
herramientas, 255
incluido, 283
líneas de guía para comprar, 201-204
mercado del, 201
Subdirectorios, 77-78
cambio, 122-124
construcción de un árbol con, 162-164
creación, 120-122
definición, 160-161
encontrando ficheros en, 172-174
reglas de uso, 165-168
suprimiento vacíos, 124-126, 170-172
visualización, 126-128

Tecla BREAK, 51, 112-113

Trazadores de gráficos, 230
Tubos, 178-181

Unidad central de procesamiento (o de tratamiento) (CPU), 2, 279-280
Unidades de cinta, 188-189
Unidades de disco:
por defecto, 19-21
función de las, 7-8

Versiones:
diferencias entre, 273-275
formateando discos para diferentes, 81
lanzamiento de diferentes, 13
órdenes en diferentes, 46
visualizando número de, 108-109

OTRAS OBRAS DE INTERES PUBLICADAS POR OSBORNE/McGRAW-HILL

CASTLEWITZ: *Introducción al Visicalc.*
ETTLIN: *Introducción al Wordstar.*
FLAST: *54 aplicaciones del Visicalc.*
FOX: *Basic básico. Guía para principiantes.*
HEILBORN: *Commodore 64. Guía del usuario.*
HEILBORN: *Programas para ciencias e ingeniería.* Edición Apple II.
HEILBORN: *VIC 20. Guía del usuario.*
HOGAN: *Sistema operativo CP/M. Guía del usuario* (2.ª ed.).
JEFFRIES: *Commodore 64. Pasatiempos y juegos.*
MOTTOLA: *Programación en lenguaje ensamblador para el Apple II.*
OSBORNE: *Guía del comprador de sistemas de gestión.*
OSBORNE: *Guía del ordenador personal PET/CBM.*
POOLE: *Algunos programas de uso común en Basic.*
POOLE: *Algunos programas de uso común en Basic.* Edición Apple II.
POOLE: *Algunos programas de uso común en Basic.* Edición Atari.
POOLE: *Algunos programas de uso común en Basic.* Edición IBM.
POOLE: *Algunos programas de uso común en Basic.* Edición PET/CBM.
POOLE: *Algunos programas de uso común en Basic.* Edición TRS-80.
POOLE: *Algunos programas de uso común en Pascal.*
POOLE: *Apple II. Guía del usuario.*
POOLE: *Programas prácticos en Basic.*
POOLE: *Programas prácticos en Basic.* Edición Apple II.
POOLE: *Programas prácticos en Basic.* Edición IBM.
POOLE: *Programas prácticos en Basic.* Edición TRS-80.
POOLE: *Programas prácticos en Pascal.*
THOMAS: *Sistema operativo UNIX. Guía del usuario.*
WAITE: *Introducción al procesamiento de palabras.*

DISCOGUIAS PUBLICADOS POR OSBORNE/McGRAW-HILL

GIFFORD: *Discoguía para Apple II.*
INGRAHAM: *Discoguía para CP/M.*
TAYLOR: *Discoguía para Atari 400/800.*
WILSON: *Discoguía para IBM/PC.*
WILSON: *Discoguía para Visicalc.*

OTRAS OBRAS DE INTERES PUBLICADAS POR BYTE-BOOKS/McGRAW-HILL

ABELSON: *Apple Logo.*
BOWLES: *Introducción al UCSD Pascal.*
CIARCIA: *Construya una computadora basada en el Z-80 (Guía de diseño y funcionamiento).*
GABY: *Gosubs. 100 subrutinas de uso común para el ZX-81 (TS 1000).*
KAMINS: *Usted y la microcomputadora. (Una introducción humanizada a la microinformática.)*
LEWART: *Programas de ciencias e ingeniería para microcomputadoras Sinclair ZX-81 compatibles con el ZX Spectrum.*

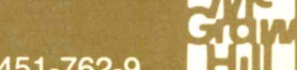
MORGAN: *Introducción al microprocesador 8086/8088 (16 bit).*
MULLISH: *Sinclair ZX-81 (TS 1000). Guía del usuario.*
NANCE: *Guía para seleccionar y adquirir su microcomputador.*
PECKHAM: *Basic para Apple II. Manual práctico.*
PECKHAM: *Basic para Commodore 64. Manual práctico.*
PECKHAM: *Basic para IBM. Manual práctico.*
PECKHAM: *Basic para TRS-80 color. Manual práctico.*
SIKONOWIZ: *Introducción al IBM/PC.*
WATT: *Aprendiendo con Logo.*

OTRAS OBRAS DE INTERES PUBLICADAS POR McGRAW-HILL

BISHOP: *ZX Spectrum (TS 2068). Teoría y proyectos de interfase.*
GOSLING: *Programación estructurada.*
HURLEY: *Introducción a la programación ZX-81 (TS 1000).*
HURLEY: *ZX Spectrum (TS 2068). Introducción al procesamiento de textos.*
PHILLIPS: *Programando el Dragón. Juegos y gráficos.*
STREET: *ZX Spectrum (TS 2068). Técnicas de procesamiento de la información.*
WILLIAMS: *ZX Spectrum (TS 2068). Diseño y programación de juegos.*
WOODS: *ZX Spectrum (TS 2068). Programación en lenguaje ensamblador.*

OTRAS OBRAS DE INTERES PUBLICADAS SOBRE EL IBM/PC

GRAHAM: IBM/PC. Guía del usuario
HOGAN: Sistema Operativo CP/M. Guía del usuario (2.ª ed.)
INGRAHAN: Discoguía para CP/M
PECKHAM: BASIC para IBM. Manual práctico
POOLE: Algunos programas de uso común en BASIC. Edición IBM/PC
POOLE: Programas prácticos en BASIC. Edición IBM/PC
SIKONOWIZ: Introducción al IBM/PC
THOMAS: Sistema Operativo UNIX. Guía del usuario



ISBN: 968-451-762-9

SISTEMA OPERATIVO MS-DOS Guía del usuario

Hoffman-Nicoloff